

VBA-Modul für PMAktuar (PMAVBA)

Dokumentation aller Funktionen/Prozeduren



Einleitung

Diese Dokumentation beschreibt die verfügbaren Funktionen und Prozeduren des VBA-Moduls (PMAVBA) für PMAktuar zum Zugriff auf den Mathematischen Modul (PMAMM) des PMAktuar-Systems. Neben den direkten Funktionen/Prozeduren der Bibliothek PMA2006.DLL sind dem Modul eine Reihe weiterer Zusatzfunktionen zum effizienten Umgang innerhalb von Microsoft Excel hinzugefügt.

Dokumentationsversion: 0.4

Modul-Header

VBA-Import-Modul zur Anbindung des Mathematischen Moduls PMA2006.DLL

Version: 1.2

Autor: Prof. Dr. Philipp Schade

Copyright: Copyright (c) 2006-2010, Prof. Dr. Philipp Schade

HINWEIS: Eigentum der MSS GmbH & Co. KG, Dortmund, Weitergabe und unlicenzierte Verwendung ist strengstens untersagt!

Gültige Lizenz über Komponenten des PMAktuar-Systems

Die Lauffähigkeit des VBA-Moduls ist nur gewährleistet, sofern auf dem Zielsystem eine gültige Installation einer lizenzierten Version von Komponenten des PMAktuar-Systems vorhanden ist. Zudem darf die Laufzeit der mathematischen Bibliothek PMA2006.DLL nicht überschritten sein.

Benötigte Dateien

Zur Verwendung des VBA-Moduls und damit zum Aufruf der Funktionen des PMAktuar-Systems ist das Vorhandensein der folgenden Dateien zwingend erforderlich:

- PMA2006_VBA.XLA
- PMA2006.DLL
- PMALiz2.DLL
- BORLANDMM.DLL
- RTL90.BPL
- VCL90.BPL

Zur Ausführung und zur Benutzung des PMAktuar VBA-Moduls ist es notwendig, dass die angegebenen Dateien im aktuellen Arbeitsverzeichnis oder im aktuell gültigen Pfad vorhanden sind. Andernfalls ist eine Ausführung nicht möglich und führt in schlimmsten Fällen zu Abstürzen von Excel. Üblicherweise genügt es, die angesprochenen Dateien im Windows\System32-Verzeichnis zur Verfügung zu stellen.

Die zentrale Schnittstelle PMA2006_VBA.XLA zu den Bibliotheken des PMAktuar-Rechenkerns ist als Add-In für Microsoft Excel konzipiert und befindet sich per Installation im Ordner C:\Programme\Gemeinsame Dateien\Math.SpezSoft\PMAktuar.

Inhaltsübersicht

Inhaltsübersicht	2
Informationsfunktionen	4
GetLizenzString-Funktion	4
GetLizenzNummer-Funktion	4
GetVersionString-Funktion	5
GetIsTestVersion-Funktion	5
Initialisierungsfunktionen	5
GetTafelCopyright-Funktion.....	5
ClearTafel-Funktion	5
ImportTafelSpez-Funktion.....	5
ImportTafel-Funktion	6
ImportGenerationstafeln-Funktion.....	6
InitHeubeckTafel1983-Funktion	6
InitHeubeckTafel1998-Funktion	7
InitHeubeckTafel2005-Funktion	7
SetHeubeck98TafelOn-Funktion	7
SetGenerationstafel-Funktion	7
GetGenerationstafelGeburtsjahr-Funktion.....	7
GetGenerationstafelGeburtsjahrPartner-Funktion.....	8
GetTafelBeginnAlter-Funktion.....	8
GetTafelEndAlter-Funktion	8
SetTafelBeginnAlter-Prozedur	8
SetTafelEndAlter-Prozedur	8
GetAbzugsterm-Funktion.....	8
InitGenauigkeit-Funktion	9
SetMindestEintrittsalter-Funktion	9
GetVerteilungsbetrag-Funktion	9
GetUebergangsjahrMehrbetrag-Funktion.....	9
SetBarwertAnzahlDezimal-Funktion.....	10
SetBigMatrix-Funktion	10
SetTeilwertMatrixRelatedBilanzjahr-Funktion	10
SetTeilwertVerteilung-Funktion	11
SetVermUebertrag-Funktion.....	11
SetIRSprungImPA-Funktion	11
SetUebergangsgeldAuffuellbetrag-Funktion.....	12
SetRiesterRegelung2001-Funktion	12
SetPlausiCheckLevel-Funktion	12
Funktionen zur Vektormodifikation.....	13
GetLeistungsvektor-Funktion	13

GetTechnicalVektor-Funktion	13
InitZinsVektor-Funktion	13
InitLeistungsvektor-Funktion	14
DoppleLeistungsvektor-Funktion.....	14
SetZeroVektor-Funktion	14
SetDynamikLeistungsvektor-Funktion	14
SetDynamikLeistungsvektorProzent-Funktion	15
SetDynamikLeistungsvektorAdditiv-Funktion	15
SetDynamikLeistungsvektorMinMax-Funktion	16
SetDynamikLeistungsvektorMax-Funktion	16
SetDynamikLeistungsvektorProzentMax-Funktion.....	16
SetDynamikLeistungsvektorAdditivMax-Funktion	17
SetLeistungsvektorMinMax-Funktion	17
SetVektorValues-Funktion	17
Barwert-/Teilwert-Berechnungsfunktionen.....	17
GetVersTechAlterByDate-Funktion	17
GetVersTechAlterByString-Funktion	18
GetEABilanzjahrbeginnJahreszahlByDate-Funktion	18
GetEABilanzjahrbeginnJahreszahlByString-Funktion	18
GetMatrixBilanzjahrInkrementByDate-Funktion.....	19
GetMatrixBilanzjahrInkrementByString-Funktion	19
GetSingleBarwert-Funktion	19
GetSingleAIWBarwert-Funktion	20
GetTW-Funktion	20
GetTWe-Funktion.....	20
GetTW2-Funktion.....	21
GetTW2004-Funktion.....	21
GetTW2005-Funktion.....	22
NEU: GetTWmod-Funktion	25
NEU: GetDBO-Funktion	28
NEU: GetPUC-Funktion.....	32
NEU: Unverfallbar2007-Funktion.....	35
Funktionen zum Auslesen der Ergebnismatrix.....	36
GetHeaderMatrixPointer-Funktion.....	36
GetErgebnisMatrixPointer-Funktion.....	36
GetRowCount-Funktion	37
GetColCount-Funktion.....	37
GetColumnTitleArray-Funktion.....	37
GetColumnTitles-Funktion	37
GetColumnTitle-Funktion	37
GetColumnNumber-Funktion	38
GetMatrixAsArray-Funktion	38
GetMatrixElement-Funktion	38

GetMatrixColumn-Funktion	38
GetMatrixColumnPart-Funktion.....	39
GetMatrixColumnPartByAge-Funktion	39
GetMatrixColumnByTitle-Funktion.....	40
Prozeduren zur Ausgabe in Tabellenblätter.....	40
PutMatrixToWorksheet-Prozedur	40
PutHeaderToWorksheet-Prozedur	40
PutMatrixColumnToWorksheet-Prozedur.....	40
PutMatrixColumnPartToWorksheetByRow-Prozedur	41
PutMatrixColumnPartToWorksheetByAge-Prozedur	41
Sonstige Berechnungsfunktionen.....	41
GetSVRente-Funktion	41
GetSVRente1998-Funktion.....	41
Grundlegende Parameter	42
Personenstatus	42
Typ der Invalidenleistung (ILTyp)	42
Art der Zusage (ZusageArt):	43
Zahlweise der Leistungen	43
Typ der Versorgungszusage (ZusageForm)	43
Finanzierungsform der Versorgungszusage (FinanzForm).....	44

Informationsfunktionen

GetLizenzString-Funktion

Syntax: `Public Function GetLizenzString() As String`

Beschreibung: Liefert den kompletten Lizenzstring der der Rechenbibliothek zugrunde liegenden Lizenzbibliothek PMALiz2.DLL entsprechend des Lizenzvertrages.

Rückgabewert: String/Zeichenkette

GetLizenzNummer-Funktion

Syntax: `Public Function GetLizenzNummer() As Long`

Beschreibung: Liefert die Lizenznummer aus der der Rechenbibliothek zugrunde liegenden Lizenzbibliothek PMALiz2.DLL entsprechend des Lizenzvertrages.

Rückgabewert: Long/Ganzzahl

GetVersionString-Funktion

Syntax: `Public Function GetVersionString() As String`

Beschreibung: Liefert die genau Versionsbezeichnung der Rechenbibliothek mit Versionsnummer und Versionsdatum.

Rückgabewert: String/Zeichenkette

GetIsTestVersion-Funktion

Syntax: `Public Function GetIsTestVersion() As Boolean`

Beschreibung: Liefert zurück, ob es sich bei der Bibliothek um eine eingeschränkte Testversion handelt oder nicht.

Rückgabewert: Boolean/Wahrheitswert mit den Möglichkeiten TRUE, falls die Bibliothek eine eingeschränkte Testversion ist und FALSE, falls nicht.

Initialisierungsfunktionen

GetTafelCopyright-Funktion

Syntax: `Public Function GetTafelCopyright() As String`

Beschreibung: (Beschreibung folgt...)

Rückgabewert: String/Zeichenkette

ClearTafel-Funktion

Syntax: `Public Function ClearTafel() As String`

Beschreibung: (Beschreibung folgt...)

Rückgabewert: String/Zeichenkette

ImportTafelSpez-Funktion

Syntax: `Public Function ImportTafelSpez(ByVal FileName As String, ByVal Trenn As String, ByVal Sex As Boolean, ByVal CopyrightLine As Integer, ByVal FirstDataLine As Integer, ByVal MessageOn As Boolean) As Boolean`

Beschreibung: (Beschreibung folgt...)

Rückgabewert: String

ImportTafel-Funktion

Syntax: Public Function ImportTafel(ByVal FileName As String, ByVal Sex As Boolean) As Boolean

Beschreibung: Die Funktion ImportTafel ist speziell konzipiert für das Einlesen und Bereitstellen eigener Ausscheideordnungen. Dabei muss die jeweilige Tafel mit den Grundwahrscheinlichkeiten als Periodentafeln (vgl. Heubeck-Richttafeln RT1983 und RT1998) vorliegen. Der Parameter FileName gibt hierbei den Dateinamen der zu importierenden und einzulesenden Datei an. Mit dem Parameter Sex wird das Geschlecht zu Grunde liegenden Person angegeben.

Hinweis: Natürlich können mit ImportTafel auch die den Heubeck-Richttafeln 1983 und 1998 beigefügten Dateien (z. B. RT98MANN.1, RT98MANN.2, RT98FRAU.1 und RT98FRAU.2) mit den Grundwahrscheinlichkeiten eingelesen werden. Dabei wird als Listentrennzeichen das Semikolon „;“ gewählt, als Zeile des Copyrighttextes Zeilennummer 6 und als erste Datenzeile Zeilennummer 8 verlangt. ImportTafel ruft intern ImportTafelSpez mit den beschriebenen Vorgaben für das Dateiformat auf.

Rückgabewert: Boolean/Wahrheitswert (True/False) mit der Angabe, ob der Import der Tafel fehlerfrei gelungen ist.

ImportGenerationstafeln-Funktion

Syntax: Public Function ImportGenerationstafeln(ByVal RTPfad As String, ByVal MessOn As Boolean) As Integer

Beschreibung: Die Funktion ImportGenerationstafeln ist speziell für den Einsatz der Generationstafeln der Heubeck-Richttafeln RT2005G konzipiert. Sie liest nacheinander alle für die Verarbeitung der Generationstafeln vorhandenen Dateien aus dem durch den Parameter RTPfad (z. B. „C:\Heurika2\Zugabe\“) vorgegebenen Verzeichnis in ein und ermöglicht dem Anwender nach Festlegung der gewünschten Tafel durch den Geburtsjahrgang (vgl. Funktionen ...), entsprechende Wahrscheinlichkeiten der geladenen Generationstafeln für die Berechnung zu benutzen. Eine im angegebenen Pfad befindliche Datei enthält dabei die Grundwahrscheinlichkeiten für den jeweiligen Geburtsjahrgang mit den Angaben der biometrischen Grundwerte x , $qxgr$, qxi , qxa , ix , hx , $y(x)$, qxw für Männer des Alters x bzw. y , $qygr$, qyi , qya , iy , hy , $x(y)$, qyw für Frauen des Alters y .

Bei Setzung des Parameters MessOn auf True werden etwaige Fehlermeldungen und eine Statusmeldung als Meldungsfenster über den Erfolg des Imports der Generationstafeldaten ausgegeben. Bei fehlerfreiem Einlesen ist der Rückgabewert 0 (Null), andernfalls gibt der Rückgabewert die Zahl fehlerhafter Dateien an.

Hinweis: Für die Richttafeln RT2005G entspricht der Pfad zu den Tafeldateien dem auf der dem Textband zu den Richttafeln beigelegten CD-ROM befindlichen Verzeichnis „Zugabe“ oder einem vergleichbaren Ort nach Kopieren auf den Computer. Dabei sind für die Funktion ImportGenerationstafeln entscheidend die Dateien mit dem Namen „2_R05Mxxxx“ mit xxxx als jeweilige Jahreszahl des Geburtsjahrgangs (Begründung: hier liegen die biometrischen Daten schon programmiersprachengerecht aufbereitet mit Dezimalpunkt anstatt Komma als Dezimaltrennzeichen vor; als Listentrennzeichen wird das Komma verwendet).

Rückgabewert: Integer/Ganzzahl mit der Angabe der fehlerhaft eingelesenen Dateien im durch RTPfad angegebenen Verzeichnis.

InitHeubeckTafel1983-Funktion

Syntax: `Public Function InitHeubeckTafel1983(ByVal Zinssatz As Double, ByVal Aktiven-
schlussalter As Integer) As String`

Beschreibung: (Beschreibung folgt...)

Rückgabewert: String

InitHeubeckTafel1998-Funktion

Syntax: `Public Function InitHeubeckTafel1998(ByVal Zinssatz As Double, ByVal Aktiven-
schlussalter As Integer) As String`

Beschreibung: (Beschreibung folgt...)

Rückgabewert: String

InitHeubeckTafel2005-Funktion

Syntax: `Public Function InitHeubeckTafel2005() As String`

Beschreibung: (Beschreibung folgt...)

Rückgabewert: String

SetHeubeck98TafelOn-Funktion

Syntax: `Public Function SetHeubeck98TafelOn(ByVal IsHeubeck98 As Boolean, ByVal Is-
JahresBarwerte As Boolean) As String`

Beschreibung: (Beschreibung folgt...)

Rückgabewert: String/Zeichenkette

SetGenerationstafel-Funktion

Syntax: `Public Function SetGenerationstafel(ByVal GeschlechtA As Boolean, ByVal Geburts-
jahrA As Integer, ByVal GeschlechtB As Boolean, ByVal GeburtsjahrB As Integer) As
String`

Beschreibung: (Beschreibung folgt...)

Rückgabewert: String/Zeichenkette

GetGenerationstafelGeburtsjahr-Funktion

Syntax: `Public Function GetGenerationstafelGeburtsjahr() As Integer`

Beschreibung: (Beschreibung folgt...)

Rückgabewert: Integer/Ganzzahl mit der Angabe der Geburtsjahreszahl der aktuell gesetzten Generationstafel für den Versorgungsberechtigten.

GetGenerationstafelGeburtsjahrPartner-Funktion

Syntax: `Public Function GetGenerationstafelGeburtsjahrPartner() As Integer`

Beschreibung: (Beschreibung folgt...)

Rückgabewert: Integer/Ganzzahl mit der Angabe der Geburtsjahreszahl der aktuell gesetzten Generationstafel für den Partner.

GetTafelBeginnAlter-Funktion

Syntax: `Public Function GetTafelBeginnAlter() As Integer`

Beschreibung: Liefert das für die aktuellen biometrischen Rechnungsgrößen gesetzte Tafelbeginnalter. Insbesondere müssen ab diesem Alter Bestandswahrscheinlichkeiten existieren.

Rückgabewert: Integer/Ganzzahl im Bereich von 1 bis 120

GetTafelEndAlter-Funktion

Syntax: `Public Function GetTafelEndAlter() As Integer`

Beschreibung: Liefert das für die aktuellen biometrischen Rechnungsgrößen gesetzte Tafelendalter **W**. Insbesondere existieren für höhere Alter keine Bestandswahrscheinlichkeiten.

Rückgabewert: Integer/Ganzzahl im Bereich von 1 bis 120

SetTafelBeginnAlter-Prozedur

Syntax: `Public Sub SetTafelBeginnAlter(ByVal NewBeginnAlter As Integer)`

Beschreibung: Setzt für die aktuellen biometrischen Rechnungsgrößen das Tafelbeginnalter.

SetTafelEndAlter-Prozedur

Syntax: `Public Sub SetTafelEndAlter(ByVal NewEndAlter As Integer)`

Beschreibung: Setzt für die aktuellen biometrischen Rechnungsgrößen das Tafelendalter.

GetAbzugsterm-Funktion

Syntax: `Public Function GetAbzugsterm(ByVal Zins As Double, ByVal ti As Integer) As Double`

Beschreibung: Liefert das Abzugsglied bei unterjährigen Zahlungen $k(t)$ in t jährlichen Perioden für den gegebenen Zins (Double) und die Anzahl gegebener unterjähriger Zahlungsperioden t_i (Integer).

Rückgabewert: Double/Gleitkommawert entsprechend der mit der Funktion InitGenauigkeit eingestellten Genauigkeit.

InitGenauigkeit-Funktion

Syntax: `Public Function InitGenauigkeit(ByVal kA As Boolean, ByVal HBA As Boolean, ByVal QLA As Boolean) As String`

Beschreibung: Diese Funktion stellt alternativ verschiedene Berechnungsgenauigkeiten ein. Der Parameter kA (Boolean) legt dabei fest, ob die Abzugsglieder bei unterjähriger Zahlung $k(t)$ mit drei Gliedern ($kA = \text{TRUE}$) oder nur als zinsunabhängiger Term $k = (t-1)/(2t)$ berechnet werden ($kA = \text{FALSE}$). Dieser Parameter ist nur von Bedeutung bei $QLA = \text{FALSE}$. Das Parameter zur Steuerung des Heubeck-Abzugsglied HBA (Boolean) legt fest, ob bei den Mittelwertbildungen im Modell der Richttafeln 1983 das Heubeck-Abzugsglied $1/(2t)$ benutzt wird. Der letzte Parameter QLA (Boolean) für die unterjährige Genauigkeit legt fest, ob bei der Anpassung der Jahresrate an die unterjährige Zahlungsweise genau oder in der üblichen Näherung mit $k(t)$ gerechnet werden soll.

Vorgabe: Vorgabeeinstellung für das Modell der Heubeck-Richttafeln 1983 ist $kA = \text{TRUE}$, $HBA = \text{WAHR}$ und $QLA = \text{FALSE}$.

Rückgabewert: String/Zeichenkette mit einer Statusmeldung, die über den Erfolg oder Misserfolg des Aufrufs Auskunft gibt (z. B. „Genauigkeit gesetzt (Wahr, Wahr, Wahr)“).

SetMindestEintrittsalter-Funktion

Syntax: `Public Function SetMindestEintrittsalter(ByVal MinAlter As Integer) As String`

Beschreibung: (Beschreibung folgt...)

Rückgabewert: String

GetVerteilungsbetrag-Funktion

Syntax: `Public Function GetVerteilungsbetrag() As Double`

Beschreibung: (Beschreibung folgt...)

Rückgabewert: Double

GetUebergangsjahrMehrbetrag-Funktion

Syntax: `Public Function GetUebergangsjahrMehrbetrag(ByVal Alter As Integer, ByVal AblAlter As Integer, ByVal IsInvalidenTod As Boolean) As Double`

Beschreibung: Die Funktion `GetUebergangsjahrMehrbetrag` liefert den zu berücksichtigen Mehrbetrag aus dem Übergangsgeld für Hinterbliebene im ersten Jahr (Todesjahr) in Abhängigkeit davon, ob der Tod im Alter `Alter` (Word) als Invalide eingetreten ist oder nicht sowie vor oder nach dem Schlussalter `AbiAlter` (Word). Im Fall des Todes als Invaliden ist der Parameter `IsInvalidenTod` (Boolean) `TRUE` andernfalls `FALSE`.

Rückgabewert: Double/Gleitkommazahl; für eine ungültige Setzung der Parameter wird Null (0,0) zurückgegeben.

SetBarwertAnzahlDezimal-Funktion

Syntax: `Public Function SetBarwertAnzahlDezimal(ByVal BWAnzahlDezimal As Integer) As String`

Beschreibung: Diese Funktion setzt die Genauigkeit für die Darstellung der Barwerte in der „großen Ergebnismatrix“ auf `BWAnzahlDezimal` (Integer) Dezimalstellen. Falls `BWAnzahlDezimal > 21` ist, wird als Defaultwert für die Ausgabedarstellung eine Genauigkeit von sechs Dezimalstellen gesetzt.

Hinweis: Diese Einstellung hat keine Auswirkungen auf die interne Rechengenauigkeit und betrifft lediglich die Darstellung in der großen Ergebnismatrix.

Rückgabewert: String/Zeichenkette mit einer Statusmeldung, die über den Erfolg oder Misserfolg des Aufrufs Auskunft gibt (z. B. „Barwerte mit Genauigkeit von 6 Dezimalstellen“).

SetBigMatrix-Funktion

Syntax: `Public Function SetBigMatrix(ByVal IsBigM As Boolean) As String`

Beschreibung: Mit Hilfe dieser Funktion wird die Option zur gleichzeitigen Erzeugung der großen Ergebnismatrix gesetzt. Zur Sicherung der Kompatibilität mit den „einfachen“ Barwert-Funktionen, die als Ergebnis die kleine zweiseitige Ergebnismatrix liefern, aktiviert diese Funktion die zusätzliche Generierung der großen Ergebnismatrix.

Rückgabewert: String/Zeichenkette mit einer Statusmeldung, die über den Erfolg oder Misserfolg des Aufrufs Auskunft gibt (z. B. „Die Erzeugung der großen Ergebnismatrix (MakeBigM) ist ein[aus]geschaltet“).

SetTeilwertMatrixRelatedBilanzjahr-Funktion

Syntax: `Public Function SetTeilwertMatrixRelatedBilanzjahr(ByVal IsTeilwert As Boolean, ByVal EAJahr As Integer, ByVal TWk As Integer) As String`

Beschreibung: Diese Funktion setzt die Option zur Darstellung von Teilwerten in der „großen Ergebnismatrix“. Der interne Schalter (`MakeTeilwert`) in der Bibliothek `PMA2005.DLL` wird auf `IsTeilwert` (Boolean) gesetzt. Falls der Parameter `EAJahr` (Integer) als Jahreszahl des Bilanzstichtages zu Beginn des Bilanzjahres, in dem der Diensteantritt/das Zusagedatum liegt, ein gültiges Jahr (1900...2050) darstellt, wird dieses bei eingeschalteter Teilwertberechnung in der Ergebnismatrix in Spalte 3 (BJahr) übernommen und hochgezählt.

Der laufende Zähler TWk (Integer) zur Markierung der Zeile des Teilwertes des aktuellen Bilanzstichtages entspricht (falls größer als Null) der entsprechenden Zeile in der Ergebnismatrix und veranlasst in diesem Falle eine Markierung des zugehörigen Alters in Spalte 2 (EA, EAX oder EAY) durch Überschreiben mit dem Wert „999“. Dieses Vorgehen soll bei maschinellen Verfahren die Ermittlung des aktuellen Teilwertes unterstützen und auch als Markierung bei der Ausgabe dieser Matrix über Bildschirm oder Drucker dienlich sein.

Zur Vereinfachung der Berechnung der Eingabe-Parameter (EAJahr und TWk) verwenden Sie idealerweise die Funktionen GetEABilanzjahrbeginnJahreszahlByString, GetEABilanzjahrbeginnJahreszahlByDate, GetMatrixBilanzjahrInkrementByString, GetMatrixBilanzjahrInkrementByDate je nachdem, ob die Eingabedaten als String in der Form „dd.mm.yyyy“ oder als Datumstyp Date vorliegen.

Rückgabewert: String/Zeichenkette mit einer Statusmeldung, die über den Erfolg oder Misserfolg des Aufrufs Auskunft gibt (z. B. „Die Bildung von Teilwerten (MakeTeilwert) ist ab 1999 eingeschaltet“).

SetTeilwertVerteilung-Funktion

Syntax: `Public Function SetTeilwertVerteilung(ByVal IstTWVerteilung As Boolean, ByVal VerteilungBST As String) As String`

Beschreibung: (Beschreibung folgt...)

Rückgabewert: String

SetVermUebertrag-Funktion

Syntax: `Public Function SetVermUebertrag(ByVal Betrag As Double, ByVal Alter As Integer) As String`

Beschreibung: Setzt den internen Schalter für die Berücksichtigung von Vermögenswert-Übertragungen (MakeVermUebT) auf WAHR und übergibt den Betrag (Double) sowie das Alter (Integer), das als versicherungstechnisches Alter zum Beginn des Bilanzjahres, in dem die Vermögenswert-Übertragung stattgefunden hat, zu verstehen ist.

Hinweis: Falls der Parameter Alter nicht im Bereich von Tafel-Mindestalter bis Tafel-Maximalalter liegt, wird NULL zurückgegeben. In der Bibliothek PMA200x.DLL wird nach jeder Teilwertberechnung der interne Schalter MakeVermUebT eigenständig auf FALSCH zurückgesetzt, so dass die Funktion im Falle eines Vermögensübertrages vor jeder Teilwertberechnung explizit aufzurufen ist.

Rückgabewert: String/Zeichenkette mit einer Statusmeldung, die über den Erfolg oder Misserfolg des Aufrufs Auskunft gibt (z. B. „Vermögenswertübertragung von xxxx EUR im Alter xx.“).

SetIRSprungImPA-Funktion

Syntax: `Public Function SetIRSprungImPA(ByVal IsIRPASprung As Boolean) As String`

Beschreibung: Setzt den Schalter IsIRPASprung (Boolean) zum Überschreiben der gebildeten Invalidenrentenvektoren, wenn bei lebenslänglichen Invalidenrenten bei Erreichen des ursprünglichen Pensionsalters ein Sprung auf eine andere/höhere Altersrente gewünscht ist.

Häufig (z. B. in Geschäftsführer-Versorgungsplänen) wird in Zusagen mit kleinerer oder abgekürzter Leistung in Falle der Invalidität bei Erreichen des (ursprünglichen) Pensionierungsalters die gleiche Altersrente verlangt, wie bei normalem Ausscheiden durch Pensionierung des aktiven Anwärters: so genannter „Sprung auf die Altersrente im Pensionsalter“. Für diese sind folgende Regelungen und insbesondere die unterschiedliche Interpretation von relevanten Null-Werten zu berücksichtigen:

- Der gewünschte Verlauf der lebenslänglichen Invalidenrente ab dem Pensionsalter, also auch eine evtl. Dynamik und Begrenzung in Höhe oder Rentenzahldauer, ist direkt im Invaliden-Leistungsvektor (analog zur Altersleistung) abzubilden.
- Zusammenhängende Null-Werte im Invalidenleistungsvektor ab Dienst Eintrittsalter werden als Wartezeit interpretiert, in der definitionsgemäß kein Anspruch auf Invalidenleistung besteht und somit insbesondere auch nicht auf Leistungen ab Pensionierung bei vorangegangener Invalidisierung – der Sprung im PA ist hier dann gegenstandslos.
- Demgegenüber sollen anspruchsfreie Zeiten vor dem PA – gekennzeichnet durch Null-Werte im Invaliden-Leistungsvektor in Altern, die dem PA „nahe sind“ – im Sinne der „versicherungsorientierten Leistungspläne“ interpretiert werden. Da die neue Kategorie nur lebenslange Invalidenleistungen betrachtet, wird in diesem Falle also stets ein Sprung im (ursprünglichen) PA auf die dann zu berücksichtigende Leistung/Rente angenommen.

Rückgabewert: String/Zeichenkette mit einer Statusmeldung, die über den Erfolg oder Misserfolg des Aufrufs Auskunft gibt (z. B. „Eine lebenslange IR wird im PA geändert (Sprung gemäß L2).“ bzw. „Eine lebenslange IR wird im PA nicht geändert.“).

SetUebergangsgeldAuffuellbetrag-Funktion

Syntax: `Public Function SetUebergangsgeldAuffuellbetrag(ByVal AWRFillJahr1VorPA As Double, ByVal AWRFillJahr1AbPA As Double, ByVal IWRFillJahr1VorPA As Double, ByVal IWRFillJahr1AbPA As Double) As String`

Beschreibung: (Beschreibung folgt...)

Rückgabewert: String

SetRiesterRegelung2001-Funktion

Syntax: `Public Function SetRiesterRegelung2001(ByVal IsRiesterMod2001 As Boolean) As String`

Beschreibung:

Rückgabewert: String

SetPlausiCheckLevel-Funktion

Syntax: `Public Function SetPlausiCheckLevel(ByVal IsHardRes As Boolean) As String`

Beschreibung: (Beschreibung folgt...)

Rückgabewert: String

Funktionen zur Vektormodifikation

GetLeistungsvektor-Funktion

Syntax: Public Function GetLeistungsvektor(ByVal LNr As Integer, ByVal VonAlter As Integer) As Variant

Beschreibung: Liefert die Werte des durch LNr (Integer) bestimmten Leistungsvektors L0, L1,..., L4 als VBA-Array zurück.

Hinweis: Beachten Sie, dass das Rückgabe-Array ein Zeilenvektor ist. Zur Darstellung als Spalte innerhalb eines Tabellenblatt ist aus diesem Grund die Excel-Funktion Application.WorksheetFunction.Transpose(...) zu verwenden. Die Anzahl der Elemente des Arrays bestimmen Sie mit den Array-Funktionen LBound(...) und UBound(...).

Rückgabewert: Variant/Enthält den eindimensionalen Array mit Gleitkommazahlen (Double) als Einträgen in Form eines Zeilenvektors.

GetTechnicalVektor-Funktion

Syntax: Public Function GetTechnicalVektor(ByVal VecName As String, ByVal VonAlter As Integer) As Variant

Beschreibung: Die Funktion liefert als Rückgabewert einen VBA-Array zur Darstellung eines der folgenden „biometrischen“ Vektoren bzw. Leistungs- oder Zinsvektoren des PMAktuar-Systems: „ZINS“, „V“, „L“, „L0“, „L1“, „L2“, „L3“, „L4“, „L5“, „L6“, „L7“, „L8“, „L9“, „IX“, „IY“, „QX“, „QY“, „QXA“, „QYA“, „QXI“, „QYI“, „HX2“, „HY2“, „YX“, „XY“, „LX“, „LY“, „LXA“, „LYA“, „LXI“, „LYI“, „QXW“, „QYW“, „HXW“, „HYW“, „LXW“, „LYW“, „LXE“, „LYE“, „LXE2“, „LYE2“, „QXE“, „QYE“, „UNVAL“, „UNVIL“, „UNVAWL“, „UNVIWL“.

Hinweis: Beachten Sie, dass der Rückgabe-Array ist ein Zeilenvektor ist. Zur Darstellung innerhalb eines Tabellenblatt ist aus diesem Grund die Funktion Application.WorksheetFunction.Transpose(...) zu verwenden. Die Anzahl der Elemente des Array bestimmen Sie mit den Array-Funktionen LBound(...) und UBound(...).

Rückgabewert: Variant/Enthält den eindimensionalen Array mit Gleitkommazahlen (Double) als Eintrag in Form eines Zeilenvektors.

InitZinsvektor-Funktion

Syntax: Public Function InitZinsvektor(ByVal Zins As Double) As String

Beschreibung: Mit InitZinsvektor initialisieren Sie den global für alle Berechnungen verwendeten Zinsvektor und in der Folge auch den Diskontierungsvektor. Ist der Parameter Zins ≥ 0 , wird der Zins konstant über alle Alter mit dem Wert Zins (Double) gesetzt. Für Zins < 0 bleibt der globale Zinsvektor unverändert und muss „von außen“ explizit gesetzt werden. In beiden Fällen werden die zugehörigen Diskontierungsfaktoren berechnet und im globalen Vektor „V“ abgelegt.

Hinweis: Zum expliziten Setzen des Zinsvektors verwenden Sie bitte die Funktion SetVektorValues mit „Zins“ oder „V“ als Name für den globalen Zins- oder Diskontierungsvektor.

Rückgabewert: String/Zeichenkette mit einer Statusmeldung, die über den Erfolg oder Misserfolg des Aufrufs Auskunft gibt (z. B. „Zinsvektor gesetzt: konstanter Diskontierungsvektor v mit Zins=5%“).

InitLeistungsvektor-Funktion

Syntax: `Public Function InitLeistungsvektor(ByVal Rate As Double, ByVal LNr As Integer) As String`

Beschreibung: Mit InitLeistungsvektor wird der durch LNr (Integer) angegebene Leistungs- bzw. Rentenvektor L0, L1,..., L4 konstant für alle Alter mit dem Gleitkommawert Rate (Double) gesetzt.

Rückgabewert: String/Zeichenkette mit einer Statusmeldung, die über den Erfolg oder Misserfolg des Aufrufs Auskunft gibt (z. B. „L1 gesetzt mit Rate 2,00“).

DoppleLeistungsvektor-Funktion

Syntax: `Public Function DoppleLeistungsvektor(ByVal DestLNr As Integer, ByVal SourceLNr As Integer, ByVal Faktor As Double) As String`

Beschreibung: Mit der Funktion DoppleLeistungsvektor wird der Ausgangsleistungs-/Rentenvektor angegeben durch den Parameter SourceLNr (Integer) in den Leistungs-/Rentenvektor DestLNr unter Verwendung des Parameters Faktor (Double) komponentenweise multiplikativ „gedoppelt“. Für jedes Alter t des neuen Zielleistungsvektors gilt damit $\text{DestLNr}(t) = \text{Faktor} * \text{SourceLNr}(t)$. Das entspricht einer Skalarmultiplikation des Ausgangsleistungsvektors SourceLNr mit dem Skalar Faktor.

Rückgabewert: String/Zeichenkette mit einer Statusmeldung, die über den Erfolg oder Misserfolg des Aufrufs Auskunft gibt (z. B. „L1 = L3 * 0,500“).

SetZeroVektor-Funktion

Syntax: `Public Function SetZeroVektor(ByVal Name As String, ByVal VonAlter As Integer, Anzahl As Integer) As String`

Beschreibung: Mit dieser Universalfunktion wird für alle Alter der Inhalt des Vektors Name (String) ab Alter VonAlter (Integer) für eine Periode von Anzahl (Integer) Jahren auf Null (0,0) gesetzt.

Hinweis: Gültige Namen für den Parameter Name sind: „ZINS“, „V“, „L“, „L0“, „L1“, „L2“, „L3“, „L4“, „L5“, „L6“, „L7“, „L8“, „L9“, „IX“, „IY“, „QX“, „QY“, „QXA“, „QYA“, „QXI“, „QYI“, „HX2“, „HY2“, „YX“, „XY“, „LX“, „LY“, „LXA“, „LYA“, „LXI“, „LYI“, „QXW“, „QYW“, „HXW“, „HYW“, „LXW“, „LYW“, „LXE“, „LYE“, „LXE2“, „LYE2“, „QXE“, „QYE“, „UNVAL“, „UNVIL“, „UNVAWL“, „UNVIWL“.

Rückgabewert: String/Zeichenkette mit einer Statusmeldung, die über den Erfolg oder Misserfolg des Aufrufs Auskunft gibt (z. B. „Vektor L1 von Alter 30 über 10 Jahre ausgenullt“).

SetDynamikLeistungsvektor-Funktion

Syntax: Public Function SetDynamikLeistungsvektor(ByVal LNr As Integer, ByVal VonAlter As Integer, ByVal AnzJahre As Integer, ByVal Startwert As Double, ByVal Dynamik As Double, ByVal WieOft As Integer) As String

Beschreibung: Mit der Funktion SetDynamikLeistungsvektor wird der durch LNr (Integer) bezeichnete Leistungs-/Rentenvektor ab Alter VonAlter (Integer) über einen zusammenhängenden Zeitraum von AnzahlJahre (Integer) beginnend mit dem Gleitkommawert Startwert (Double) konstant gesetzt. Ist der Parameter Dynamik (Double) mit einem Wert größer als Null gesetzt, so wird der Startwert alle WieOft (Integer) Jahre dynamisiert. Dabei bezeichnet ein Wert Dynamik ≤ 1 , dass der Startwert prozentual mit der entsprechenden Dynamik dynamisiert werden soll, ein Wert Dynamik > 1 wird als ein absoluter Dynamikwert verstanden, der jedes Jahr in der gleichen Höhe auf den Vorjahreswert addiert wird. Diese Funktion bildet die Ausgangsfunktion für alle folgenden SetDynamikLeistungsvektorXXX-Funktionen.

Hinweis: Ein Wert AnzahlJahre = 0 wird als unbeschränkter Zeitraum verstanden, was eine Setzung des Vektors bis zum Tafelende ! zur Folge hat. Hat der Parameter Startwert den Wert Startwert = 0, so hat das nicht ein Ausnullen des Vektors AnzahlJahre lang zur Folge, sondern erzwingt als Startwert im Alter VonAlter, den an dieser Stelle schon im Vektor durch eventuelle vorherige Setzungen vorhandenen Wert zu benutzen. Auf diese Weise kann ein Vektor beliebig aufgebaut werden und ermöglicht es zum Beispiel, ohne Kenntnis des konkreten Vektorwertes im Alter VonAlter eine dort beginnende Dynamik „aufzusetzen“.

Rückgabewert: String/Zeichenkette mit einer Statusmeldung, die über den Erfolg oder Misserfolg des Aufrufs Auskunft gibt (z. B. „L1 mit Dynamik ab Alter 50“).

SetDynamikLeistungsvektorProzent-Funktion

Syntax: Public Function SetDynamikLeistungsvektorProzent(ByVal LNr As Integer, ByVal VonAlter As Integer, ByVal AnzJahre As Integer, ByVal Startwert As Double, ByVal DynProzent As Double, ByVal WieOft As Integer) As String

Beschreibung: SetDynamikLeistungsvektorProzent arbeitet analog zur Ausgangsfunktion SetDynamikLeistungsvektor mit dem Unterschied, dass die Angabe für die Dynamik hier nur prozentual verstanden wird und eine Dynamisierung des Startwerts ab VonAlter alle WieOft Jahre zur Folge hat.

Hinweis: Im Übrigen sind hier auch negative Dynamiken zulässig, die in analoger Weise zu prozentual abnehmenden Werten im Leistungs-/Rentenvektor führen.

Rückgabewert: String/Zeichenkette mit einer Statusmeldung, die über den Erfolg oder Misserfolg des Aufrufs Auskunft gibt (z. B. „L1 mit Dynamik ab Alter 20“).

SetDynamikLeistungsvektorAdditiv-Funktion

Syntax: Public Function SetDynamikLeistungsvektorAdditiv(ByVal LNr As Integer, ByVal VonAlter As Integer, ByVal AnzJahre As Integer, ByVal Startwert As Double, ByVal DynAdditiv As Double, ByVal WieOft As Integer) As String

Beschreibung: SetDynamikLeistungsvektorAdditiv arbeitet analog zur Ausgangsfunktion SetDynamikLeistungsvektor mit dem Unterschied, dass die Angabe für den Parameter Dynamik hier nur als absolute additive Größe verstanden wird und eine Dynamisierung des Startwerts ab VonAlter alle WieOft Jahre sich als jährlicher konstanter Zuwachs ausdrückt.

Hinweis: Im Übrigen sind hier auch negative Dynamiken zulässig, die in analoger Weise zu jährlich in gleicher Höhe abnehmenden Werten im Leistungs-/Rentenvektor führen.

Rückgabewert: String/Zeichenkette mit einer Statusmeldung, die über den Erfolg oder Misserfolg des Aufrufs Auskunft gibt (z. B. „L1 mit Dynamik ab Alter 20“).

SetDynamikLeistungsvektorMinMax-Funktion

Syntax: `Public Function SetDynamikLeistungsvektorMinMax(ByVal LNr As Integer, ByVal VonAlter As Integer, ByVal AnzJahre As Integer, ByVal Startwert As Double, ByVal Dynamik As Double, ByVal WieOft As Integer, ByVal MinWert As Double, ByVal MaxWert As Double) As String`

Beschreibung: Diese Funktion verhält sich identisch zu der Ausgangsfunktion SetDynamikLeistungsvektor. Die beiden zusätzlichen Parameter erlauben eine Beschränkung der Werte des Vektors durch MaxWert (Double) nach oben und durch MinWert > 0 (Double) nach unten. Dabei werden zunächst diejenigen Werte, die durch die Dynamisierung den Wert MaxWert überschreiten durch MaxWert ersetzt und anschließend alle Werte des Vektors mit einem Eintrag ungleich Null – sofern kleiner – auf den Wert MinWert angehoben.

Rückgabewert: String/Zeichenkette mit einer Statusmeldung, die über den Erfolg oder Misserfolg des Aufrufs Auskunft gibt (z. B. „L1 mit Dynamik ab Alter 20, höchstens 200.000,00, mindestens 84.000,00“).

SetDynamikLeistungsvektorMax-Funktion

Syntax: `Public Function SetDynamikLeistungsvektorMax(ByVal LNr As Integer, ByVal VonAlter As Integer, ByVal AnzJahre As Integer, ByVal Startwert As Double, ByVal Dynamik As Double, ByVal WieOft As Integer, ByVal MaxWert As Double) As String`

Beschreibung: Diese Funktion verhält sich identisch zu der Ausgangsfunktion SetDynamikLeistungsvektor. Der zusätzliche Parameter MaxWert (Double) erlaubt eine Beschränkung der Werte des Vektors durch MaxWert nach oben, d. h. die Dynamik wird nur bis zu einer Höchstgrenze berücksichtigt, danach wird der Vektor in Höhe von MaxWert konstant fortgeführt.

Rückgabewert: String/Zeichenkette mit einer Statusmeldung, die über den Erfolg oder Misserfolg des Aufrufs Auskunft gibt (z. B. „L1 mit Dynamik ab Alter 20, höchstens 200.000,00“).

SetDynamikLeistungsvektorProzentMax-Funktion

Syntax: `Public Function SetDynamikLeistungsvektorProzentMax(ByVal LNr As Integer, ByVal VonAlter As Integer, ByVal AnzJahre As Integer, ByVal Startwert As Double, ByVal DynProzent As Double, ByVal WieOft As Integer, ByVal MaxWert As Double) As String`

Beschreibung: Diese Funktion verhält sich identisch zu der Funktion SetDynamikLeistungsvektorProzent. Der zusätzliche Parameter MaxWert (Double) erlaubt eine Beschränkung der Werte des Vektors durch MaxWert nach oben, d. h. die Dynamik wird nur bis zu einer Höchstgrenze berücksichtigt, danach wird der Vektor in Höhe von MaxWert konstant fortgeführt.

Rückgabewert: String/Zeichenkette mit einer Statusmeldung, die über den Erfolg oder Misserfolg des Aufrufs Auskunft gibt (z. B. „L1 mit Dynamik ab Alter 20, höchstens 200.000,00“).

SetDynamikLeistungsvektorAdditivMax-Funktion

Syntax: Public Function SetDynamikLeistungsvektorAdditivMax(ByVal LNr As Integer, ByVal VonAlter As Integer, ByVal AnzJahre As Integer, ByVal Startwert As Double, ByVal DynAdditiv As Double, ByVal WieOft As Integer, ByVal MaxWert As Double) As String

Beschreibung: Diese Funktion verhält sich identisch zu der Funktion SetDynamikLeistungsvektorAdditiv. Der zusätzliche Parameter MaxWert (Double) erlaubt eine Beschränkung der Werte des Vektors durch MaxWert nach oben, d. h. die Dynamik wird nur bis zu einer Höchstgrenze berücksichtigt, danach wird der Vektor in Höhe von MaxWert konstant fortgeführt.

Rückgabewert: String/Zeichenkette mit einer Statusmeldung, die über den Erfolg oder Misserfolg des Aufrufs Auskunft gibt (z. B. „L1 mit Dynamik ab Alter 20, höchstens 200.000,00“).

SetLeistungsvektorMinMax-Funktion

Syntax: Public Function SetLeistungsvektorMinMax(ByVal LNr As Integer, ByVal MinWert As Double, ByVal MaxWert As Double) As String

Beschreibung: Die Funktion SetLeistungsvektorMinMax beschränkt im durch LNr (Integer) angegebenen Leistungs-/Rentenvektor alle Werte ungleich Null durch MinWert nach unten, falls der Parameter MinWert (Double) größer als Null ist und durch MaxWert nach oben, falls der Parameter MaxWert größer als Null ist.

Rückgabewert: String/Zeichenkette mit einer Statusmeldung, die über den Erfolg oder Misserfolg des Aufrufs Auskunft gibt (z. B. „Leistungsvektor L1 beschränkt, mindestens 1200,00, höchstens 1800,00“).

SetVektorValues-Funktion

Syntax: Public Function SetVektorValues(ByVal Name As String, ByVal VonAlter As Integer, ByVal Q As Long) As String

Beschreibung: (Beschreibung folgt...)

Rückgabewert: String

Barwert-/Teilwert-Berechnungsfunktionen

GetVersTechAlterByDate-Funktion

Syntax: Public Function GetVersTechAlterByDate(ByVal GebDatum As Date, ByVal RefDatum As Date, ByVal BilanzStichtagDatum As Date) As Integer

Beschreibung: Diese Funktion liefert das versicherungstechnische Eintrittsalter zu Beginn des Bilanzjahres, in das der Dienst Eintritt/Zusagedatum fällt. Dabei legt der Parameter RefDatum (Date) das entsprechende Datum fest, d. h. entweder der Dienst Eintritt zur Bestimmung des Eintrittsalters zu Beginn des Bilanzjahres, in das der Dienst Eintritt fällt oder das Zusagedatum zur Bestimmung des Zusagealters zu Beginn des Bilanzjahres der durch das Zusagedatum bestimmten Zusage. Alle Parameter werden als

Datum vom Typ Date erwartet. Rückgabewert ist das versicherungstechnische Eintrittsalter zu Beginn des jeweiligen Bilanzjahres. Der Parameter GebDatum (Date) bezeichnet hierbei das Geburtsdatum, der Parameter BilanzStichtagDatum (Date) den genauen aktuellen Bilanzstichtag.

Rückgabewert: Integer/Ganzzahl mit dem zugehörigen versicherungstechnischen Eintrittsalter am Beginn des Bilanzjahres des Dienst Eintritts oder der Zusage.

GetVersTechAlterByString-Funktion

Syntax: `Public Function GetVersTechAlterByString(ByVal GebDatum As String, ByVal RefDatum As String, ByVal BilanzStichtagDatum As String) As Integer`

Beschreibung: Diese Funktion liefert das versicherungstechnische Eintrittsalter zu Beginn des Bilanzjahres, in das der Dienst Eintritt/Zusagedatum fällt. Dabei legt der Parameter RefDatumStr (String) das entsprechende Datum fest, d. h. entweder der Dienst Eintritt zur Bestimmung des Eintrittsalters zu Beginn des Bilanzjahres, in das der Dienst Eintritt fällt oder das Zusagedatum zur Bestimmung des Zusagealters zu Beginn des Bilanzjahres der durch das Zusagedatum bestimmten Zusage. Alle Parameter werden als String in der Form "dd.mm.yyyy" erwartet. Rückgabewert ist das versicherungstechnische Eintrittsalter zu Beginn des jeweiligen Bilanzjahres. Der Parameter GebDatumStr (String) bezeichnet hierbei das Geburtsdatum, der Parameter BilanzStichtagDatumStr (String) den genauen aktuellen Bilanzstichtag.

Rückgabewert: Integer/Ganzzahl mit dem zugehörigen versicherungstechnischen Eintrittsalter am Beginn des Bilanzjahres des Dienst Eintritts oder der Zusage.

GetEABilanzjahrbeginnJahreszahlByDate-Funktion

Syntax: `Public Function GetEABilanzjahrbeginnJahreszahlByDate(ByVal GebDatum As Date, ByVal RefDatum As Date, ByVal BilanzStichtagDatum As Date) As Integer`

Beschreibung: Diese Funktion liefert die Jahreszahl des Bilanzstichtages am Beginn des Bilanzjahres, in das das durch RefDatum (Date) bezeichnete Ereignis zeigt (Dienst Eintritt/Zusagebeginn). Diese Jahreszahl ist damit identisch mit der Jahreszahl des vorangegangenen Bilanzjahres. Beispiel: RefDatum = Dienst Eintritt = 01.05.2005, damit Beginn des Bilanzjahres in das der Dienst Eintritt fällt 01.01.2005, damit unmittelbar vorangehender Bilanzstichtag 31.12.2004, damit Jahreszahl 2004. Alle Parameter werden vom Typ Date erwartet. Der Parameter GebDatum (Date) bezeichnet hierbei das Geburtsdatum, der Parameter BilanzStichtagDatum (Date) den genauen aktuellen Bilanzstichtag.

Rückgabewert: Integer/Ganzzahl mit der Jahreszahl des angesprochenen Bilanzstichtags.

GetEABilanzjahrbeginnJahreszahlByString-Funktion

Syntax: `Public Function GetEABilanzjahrbeginnJahreszahlByString(ByVal GebDatum As String, ByVal RefDatum As String, ByVal BilanzStichtagDatum As String) As Integer`

Beschreibung: Diese Funktion liefert die Jahreszahl des Bilanzstichtages am Beginn des Bilanzjahres, in das das durch RefDatum (String) bezeichnete Ereignis zeigt (Dienst Eintritt/Zusagebeginn). Diese Jahreszahl ist damit identisch mit der Jahreszahl des vorangegangenen Bilanzjahres. Beispiel: RefDatum = Dienst Eintritt = 01.05.2005, damit Beginn des Bilanzjahres in das der Dienst Eintritt fällt 01.01.2005, damit unmittelbar vorangehender Bilanzstichtag 31.12.2004, damit Jahreszahl 2004. Alle

Parameter werden als String der Form „dd.mm.yyyy“ erwartet. Der Parameter GebDatum (String) bezeichnet hierbei das Geburtsdatum, der Parameter BilanzStichtagDatum (String) den genauen aktuellen Bilanzstichtag.

Rückgabewert: Integer/Ganzzahl mit der Jahreszahl des angesprochenen Bilanzstichtags.

GetMatrixBilanzjahrInkrementByDate-Funktion

Syntax: `Public Function GetMatrixBilanzjahrInkrementByDate(ByVal GebDatum As Date, ByVal RefDatum As Date, ByVal BilanzStichtagDatum As Date) As Integer`

Beschreibung: Diese Funktion bestimmt für die große Ergebnismatrix die zu markierende Zeile mit der Jahreszahl des aktuellen Bilanzjahres. Die durch den Rückgabewert ermittelte Zeile wird in der Spalte „EA[X|Y]“ durch den Eintrag „999“ markiert. Damit die Markierung sichtbar wird und eine Spalte mit den Jahreszahlen der jeweiligen Bilanzstichtage aufgenommen wird, muss mit dem hier ermittelten Rückgabewert und dem Rückgabewert der Funktion GetEABilanzjahrbeginnJahreszahlBy[String|Date] die Funktion SetTeilwertMatrixRelatedBilanzjahr aufgerufen werden. Alle Parameter werden hier als Datum vom Typ Date erwartet. Der Parameter GebDatum (Date) bezeichnet hierbei das Geburtsdatum, der Parameter BilanzStichtagDatum (Date) den genauen aktuellen Bilanzstichtag. Mit dem Parameter RefDatum (Date) wird das zugehörige Referenzdatum, also der Diensteintritt oder das Zusagedatum angegeben.

Rückgabewert: Integer/Ganzzahl, die das Zeileninkrement für die aktuelle Bilanzjahreszeile angibt.

GetMatrixBilanzjahrInkrementByString-Funktion

Syntax: `Public Function GetMatrixBilanzjahrInkrementByString(ByVal GebDatum As String, ByVal RefDatum As String, ByVal BilanzStichtagDatum As String) As Integer`

Beschreibung: Diese Funktion bestimmt für die große Ergebnismatrix die zu markierende Zeile mit der Jahreszahl des aktuellen Bilanzjahres. Die durch den Rückgabewert ermittelte Zeile wird in der Spalte „EA[X|Y]“ durch den Eintrag „999“ markiert. Damit die Markierung sichtbar wird und eine Spalte mit den Jahreszahlen der jeweiligen Bilanzstichtage aufgenommen wird, muss mit dem hier ermittelten Rückgabewert und dem Rückgabewert der Funktion GetEABilanzjahrbeginnJahreszahlBy[String|Date] die Funktion SetTeilwertMatrixRelatedBilanzjahr aufgerufen werden. Alle Parameter werden hier als String der Form „dd.mm.yyyy“ erwartet. Der Parameter GebDatum (String) bezeichnet hierbei das Geburtsdatum, der Parameter BilanzStichtagDatum (String) den genauen aktuellen Bilanzstichtag. Mit dem Parameter RefDatum (String) wird das zugehörige Referenzdatum, also der Diensteintritt oder das Zusagedatum angegeben.

Rückgabewert: Integer/Ganzzahl, die das Zeileninkrement für die aktuelle Bilanzjahreszeile angibt.

GetSingleBarwert-Funktion

Syntax: `Public Function GetSingleBarwert(ByVal BarwertName As String, ByVal SexMann As Boolean, ByVal Alter As Integer, ByVal Alter2 As Integer, ByVal Zinssatz As Double, ByVal AblAlter As Integer, ByVal t As Integer, ByVal LNr As Integer) As Double`

Beschreibung: (Beschreibung folgt...)

Rückgabewert: Double

GetSingleAIWBarwert-Funktion

Syntax: Public Function GetSingleAIWBarwert(ByVal BarwertName As String, ByVal SexMann As Boolean, ByVal X As Integer, ByVal Y As Integer, ByVal Zinssatz As Double, ByVal AblAlter As Integer, ByVal t As Integer, ByVal LARNr As Integer, ByVal LIRNr As Integer, ByVal IRabgek As Boolean, ByVal IRDynamik As Double, ByVal IRWieOft As Integer, ByVal IRGrenze As Double, ByVal IRBisAlter As Integer, ByVal LaWRNr As Integer, ByVal aWRDynamik As Double, ByVal aWRWieOft As Integer, ByVal aWRGrenze As Double, ByVal aWRBisAlter As Integer, ByVal LiWRNr As Integer, ByVal iWRDynamik As Double, ByVal iWRWieOft As Integer, ByVal iWRGrenze As Double, ByVal iWRBisAlter As Integer) As Double

Beschreibung: (Beschreibung folgt...)

Rückgabewert: Double

GetTW-Funktion

Syntax: Public Function GetTW (ByVal Status As Integer, ByVal Alter1 As Integer, ByVal Sex1 As Boolean, ByVal Alter2 As Integer, ByVal Sex2 As Boolean, ByVal GesBestand As Boolean, ByVal ALeist As Integer, ByVal ILeist As Integer, ByVal AHLeist As Integer, ByVal IHLeist As Integer, ByVal Jahr As Integer, ByVal TWk As Integer, ByVal Zins As Double, ByVal AblAlter As Integer, ByVal t As Integer, ByVal LARNr As Integer, ByVal LIRNr As Integer, ByVal abgek As Boolean, ByVal IRDyn As Double, ByVal IRWieOft As Integer, ByVal IRGrenze As Double, ByVal IRBisAlter As Integer, ByVal LaWRNr As Integer, ByVal aWRDyn As Double, ByVal aWRWieOft As Integer, ByVal aWRGrenze As Double, ByVal aWRBisAlter As Integer, ByVal LiWRNr As Integer, ByVal iWRDyn As Double, ByVal iWRWieOft As Integer, ByVal iWRGrenze As Double, ByVal iWRBisAlter As Integer) As Long

Beschreibung: (Beschreibung folgt...)

Hinweis: Die Funktion GetTW war spezialisiert für die Ausführung innerhalb von Excel4-Makros und wird aus Gründen der Abwärtskompatibilität weiterhin aufgeführt. Aktuelle Implementierungen sollten stattdessen die Funktionen GetTW2004(...), GetTW2005(...), GetTWmod(...), GetDBO(...) bzw. GetPUC(...) verwenden (s. u.).

Rückgabewert: Long/Ganzzahl als Adresse der großen Ergebnismatrix-Struktur (PEMatM) im Speicher.

GetTWe-Funktion

Syntax: Public Function GetTWe (ByVal Status As Integer, ByVal Alter1 As Integer, ByVal Sex1 As Boolean, ByVal Alter2 As Integer, ByVal GesBestand As Boolean, ByVal ALeist As Integer, ByVal ILeist As Integer, ByVal AHLeist As Integer, ByVal IHLeist As Integer, ByVal Zins As Double, ByVal AblAlter As Integer, ByVal t As Integer, ByVal LARNr As Integer, ByVal LIRNr As Integer, ByVal abgek As Boolean, ByVal IRDyn As Double, ByVal IRWieOft As Integer, ByVal IRGrenze As Double, ByVal BisIRAlter As Integer, ByVal LaWRNr As Integer, ByVal aWRDyn As Double, ByVal aWRWieOft As Integer, ByVal aWRGrenze As Double, ByVal BisawRAlter As Integer, ByVal LiWRNr As Integer, ByVal iWRDyn As Double, ByVal iWRWieOft As Integer, ByVal iWRGrenze As Double, ByVal BisiWRAlter As Integer) As Long

Beschreibung: (Beschreibung folgt...)

Hinweis: Die Funktion GetTWe war spezialisiert für die Ausführung innerhalb von Excel4-Makros und wird aus Gründen der Abwärtskompatibilität weiterhin aufgeführt. Aktuelle Implementierungen sollten stattdessen die Funktionen GetTW2004(...), GetTW2005(...), GetTWmod(...), GetDBO(...) bzw. GetPUC(...) verwenden (s. u.).

Rückgabewert: Long/Ganzzahl als Adresse der großen Ergebnismatrix-Struktur (PEMatM) im Speicher.

GetTW2-Funktion

Syntax: Public Function GetTW2 (ByVal Status As Integer, ByVal Alter1 As Integer, ByVal Sex1 As Boolean, ByVal Alter2 As Integer, ByVal Sex2 As Boolean, ByVal GesBestand As Boolean, ByVal ALeist As Integer, ByVal lLeist As Integer, ByVal AHLeist As Integer, ByVal IHLeist As Integer, ByVal Jahr As Integer, ByVal TWk As Integer, ByVal Zins As Double, ByVal AblAlter As Integer, ByVal t As Integer, ByVal LARNr As Integer, ByVal LIRNr As Integer, ByVal abgek As Boolean, ByVal IRDyn As Double, ByVal IRWieOft As Integer, ByVal IRGrenze As Double, ByVal BisIRAlter As Integer, ByVal LaWRNr As Integer, ByVal aWRDyn As Double, ByVal aWRWieOft As Integer, ByVal aWRGrenze As Double, ByVal BisaWRAlter As Integer, ByVal LiWRNr As Integer, ByVal iWRDyn As Double, ByVal iWRWieOft As Integer, ByVal iWRGrenze As Double, ByVal BisiWRAlter As Integer) As Long

Beschreibung: (Beschreibung folgt...)

Hinweis: Die Funktion GetTW2 war spezialisiert für die Ausführung innerhalb von Excel4-Makros und wird aus Gründen der Abwärtskompatibilität weiterhin aufgeführt. Aktuelle Implementierungen sollten stattdessen die Funktionen GetTW2005(...), GetTWmod(...), GetDBO(...) bzw. GetPUC(...) verwenden (s. u.).

Rückgabewert: Long/Ganzzahl als Adresse der großen Ergebnismatrix-Struktur (PEMatM) im Speicher.

GetTW2004-Funktion

Syntax: Public Function GetTW2004 (ByVal Status As Integer, ByVal Alter1 As Integer, ByVal Sex1 As Boolean, ByVal Alter2 As Integer, ByVal Sex2 As Boolean, ByVal GesBestand As Boolean, ByVal ALeist As Integer, ByVal lLeist As Integer, ByVal AHLeist As Integer, ByVal IHLeist As Integer, ByVal Jahr As Integer, ByVal TWk As Integer, ByVal Zins As Double, ByVal AblAlter As Integer, ByVal tAR As Integer, ByVal LARNr As Integer, ByVal LIRNr As Integer, ByVal ILTyp As Integer, ByVal IRDyn As Double, ByVal IRWieOft As Integer, ByVal IRGrenze As Double, ByVal BisIRAlter As Integer, ByVal tIR As Integer, ByVal IRisKapital As Boolean, ByVal LaWRNr As Integer, ByVal aWRDyn As Double, ByVal aWRWieOft As Integer, ByVal aWRGrenze As Double, ByVal BisaWRAlter As Integer, ByVal taWR As Integer, ByVal aWRisKapital As Boolean, ByVal LiWRNr As Integer, ByVal iWRDyn As Double, ByVal iWRWieOft As Integer, ByVal iWRGrenze As Double, ByVal BisiWRAlter As Integer, ByVal tiWR As Integer, ByVal iWRisKapital As Boolean, ByVal Alter1Zus As Integer, ByVal JahrZus As Integer, ByVal AltZusagekey As Integer, ByVal ZusageForm As Integer, ByVal EUmwAusfinanziert As Boolean, ByVal UnvKey As Integer, ByVal actUnvAL As Double, ByVal actUnvIL As Double, ByVal actUnvAWL As Double, ByVal actUnvIWL As Double, ByVal actUnvZins As Double) As Long

Beschreibung: Funktion zur Bewertung einer Versorgungsverpflichtung nach dem sog. "Teilwertverfahren" (vgl. § 6a EStG) auf Basis von Periodensterbetafeln (bspw. Heubeck-Richttafeln RT 1986 bzw. 1998, nicht aber Generationensterbetafeln wie RT 2005G). Für die Ermittlung der notwendigen Barwerte der Anwartschaften auf Alters-, Invaliden- sowie Hinterbliebenenleistungen sind als Voraussetzung entsprechend die Leistungsvektoren L1, L2, L3 bzw. L4 geeignet vorzubelegen.

Hinweis: Die Funktion GetTW2004 ist insbesondere ausgerichtet auf die Ausführung im Zusammenhang mit der Berechnung alter Teilwerte auf Basis der Heubeck-Richttafeln 1998 und älter (keine Generationentafeln). Für Teilwertberechnungen auf Basis von Generationentafeln wird stattdessen verwiesen auf die Funktionen GetTW2005(...), GetTWmod(...), GetDBO(...) bzw. GetPUC(...) verwenden (s. u.).

Rückgabewert: Long/Ganzzahl als Adresse der großen Ergebnismatrix-Struktur (PEMatM) im Speicher.

GetTW2005-Funktion

Syntax: Public Function GetTW2005(ByVal Status As Integer, ByVal GebDat1 as Date, ByVal Alter1 As Integer, ByVal Sex1 As Boolean, ByVal GebDat2 as Date, ByVal Alter2 As Integer, ByVal Sex2 As Boolean, ByVal GesBestand As Boolean, ByVal ALeist As Integer, ByVal lLeist As Integer, ByVal AHLeist As Integer, ByVal IHLeist As Integer, ByVal Jahr As Integer, ByVal TWk As Integer, ByVal Zins As Double, ByVal AblAlter As Integer, ByVal t As Integer, ByVal LARNr As Integer, ByVal LIRNr As Integer, ByVal ILTyp As Integer, ByVal IRDyn As Double, ByVal IRWieOft As Integer, ByVal IRGrenze As Double, ByVal BisIRAlter As Integer, ByVal LaWRNr As Integer, ByVal aWRDyn As Double, ByVal aWRWieOft As Integer, ByVal aWRGrenze As Double, ByVal BisaWRAlter As Integer, ByVal LiWRNr As Integer, ByVal iWRDyn As Double, ByVal iWRWieOft As Integer, ByVal iWRGrenze As Double, ByVal BisiWRAlter As Integer, ByVal Alter1Zus As Integer, ByVal JahrZus As Integer, ByVal AltZusagekey As Integer, ByVal ZusageForm As Integer, ByVal EUmwAusfinanziert As Boolean, ByVal UnvKey As Integer, ByVal actUnvAL As Double, ByVal actUnvIL As Double, ByVal actUnvAWL As Double, ByVal actUnvIWL As Double, ByVal actUnvZins As Double) As Long

Beschreibung: Funktion zur Bewertung einer Versorgungsverpflichtung nach dem sog. "Teilwertverfahren" (vgl. § 6a EStG) auf Basis von Generationentafeln (bspw. Heubeck-Richttafeln RT 2005G). Für die Ermittlung der notwendigen Barwerte der Anwartschaften auf Alters-, Invaliden- sowie Hinterbliebenenleistungen sind als Voraussetzung entsprechend die Leistungsvektoren L1, L2, L3 bzw. L4 geeignet vorzublegen. Die einzelnen Funktionsparameter haben dabei die folgende Bedeutung:

- Status: Personenstatus des Versorgungsberechtigten (Anwärter, Altersrentner usw.) gemäß Übersicht über die Personenstatus (s. u.),
- GebDat/GebDat1: Geburtsdatum des Versorgungsberechtigten (Anzahl der Tage seit dem 30.12.1899 12.00 Uhr, Nachkommastellen beschreiben Uhrzeit des Tages – irrelevant für reine Datumsangaben),
- Alter/Alter1: Alter des Versorgungsberechtigten bei Statureintritt, d. h. das Datum des Dienstetrtritts für Versorgungsberechtigte mit Personenstatus „Anwärter“ oder das Datum des Leistungsfalls für Empfänger laufender Leistungen,
- Sex/Sex1: Geschlecht des Versorgungsberechtigten als boolescher Wert (False=weiblich, True=männlich),
- GebDat2/GebDatP: Geburtsdatum des Partners des Versorgungsberechtigten (Anzahl der Tage seit dem 30.12.1899 12:00 Uhr, Nachkommastellen beschreiben Uhrzeit des Tages – irrelevant für reine Datumsangaben),
- Alter2/AlterP: versicherungstechnisches Alter des Partners des Versorgungsberechtigten bei Statureintritt des Versorgungsberechtigten,
- Sex2/SexP: Geschlecht des Partners des Versorgungsberechtigten (False=weiblich, True=männlich, beachte: gleichgeschlechtliche Beziehungen sind möglich),
- GesBestand: Boolescher Wert zur Angabe, ob die Bewertung im Gesamt- oder im Aktivenbestand erfolgt (True=Gesamtbestand, False=Aktivenbestand),
- ALeist: Angabe zur Berücksichtigung einer Altersleistung (1: Leistungen ab Pensionsalter, 0: keine Leistungen; Hinweis: bei Setzung „0“ werden jegliche Werte im Altersleistungsvektor ignoriert),
- lLeist: Angabe zur Berücksichtigung einer Invalidenleistung (1 : Leistungen bei Invalidität, 0 : keine Leistungen; Hinweis: bei Setzung „0“ werden jegliche Werte im Invalidenleistungsvektor ignoriert),
- AHLeist: Angabe zur Berücksichtigung einer Hinterbliebenenleistung (2: Hinterbliebenenleistungen, 1: Leistungen nur bei Aktiventod, d. h. lediglich Aktiven-Hinterbliebenenleistungen sind vorgegeben, 0: keine Hinterbliebenenleistungen),
- IHLeist: Angabe zur Berücksichtigung einer Invalidenhinterbliebenenleistung (1: Hinterbliebenenleistungen nur bei Invalidentod, 0: keine Leistungen),

- Jahr: Jahreszahl vom Bilanzstichtag des Bilanzjahres des Diensteintritts (damit also korrespondierende Jahreszahl des Bilanzstichtages des Beginns des Bilanzjahres; Beispiel: Diensteintritt 3.6.1992, dann Jahr=1991, da Bilanzjahresbeginn am 1.1.1992 und damit identisch am 31.12.1991; Angabe ist für die Visualisierung des Äquivalenzprinzips innerhalb des Teilwertverfahrens am Beginn des Wirtschaftsjahres des Diensteintritts notwendig),
- TWk: (Zeilen-)Nummer derjenigen Zeile innerhalb der großen Ergebnismatrix, in die das aktuelle Bilanzjahr bzw. genauer der aktuelle Bilanzstichtag fällt; die Ermittlung erfolgt am einfachsten über die dafür bereitgestellte Funktion GetEABilanzjahrbeginnJahreszahlByDate(...); TWk dient zur lediglich zur Identifizierung der sog. Bilanzjahreszeile,
- Zins: für die aktuelle Bewertung relevanter Rechnungszinssatz (Vorgabe für die Steuerbilanz 6 %, Vorgabe für die Handelsbilanz nach den jeweiligen ermittelten Stichtagszinssätzen der Bundesbank gemäß RückAbzinsV),
- AblAlter: Pensionsalter des Versorgungsberechtigten,
- tAR: Schlüssel der für die Angabe der Zahlweise für die Altersleistung (vgl. dazu unten die Auflistung möglicher Zahlweisen),
- LARNr: Nummer Altersleistungsvektors (in der Regel LARNr=1),
- LIRNr: Nummer des Invalidenleistungsvektors (in der Regel LIRNr=2),
- ILTyp: Typ der Invalidenleistung gemäß Erläuterung unten (1: abgekürzte Invalidenleistung bis Pensionsalter, 0: lebenslängliche Invalidenleistung, 2: auf das Pensionsalter aufgeschobene Invalidenleistung, 3: Invalidenleistung mit Sprung im Pensionsalter; beachte bei ILTyp=2 bzw. ILTyp=3 die Sonderbehandlung des Invalidenleistungsvektors),
- IRDyn: Dynamiksatzeiner laufenden Invalidenleistung, d. h. Dynamik der Invalidenrente (Dezimalzahl, bspw. 0,02 für eine Dynamik der Rente in Höhe von 2 % bezogen auf den jeweiligen Vorjahreswert),
- IRWieOft: Parameter, der angibt, alle wie viel Jahre die Dynamisierung der Invalidenleistung erfolgt (regelmäßig ist IRWieOft=1 für jährliche Erhöhung der Renten entsprechend des Dynamiksatzes IRDyn; bspw. würde IRWieOft=3 mit IRDyn=0,03 bedeuten, dass die Invalidenrenten alle drei Jahre um 3 % angehoben werden würde),
- IRGrenze: Höchst-/Grenzwert der Rentenhöhe für die Dynamisierung der laufenden Invalidenrente (IRGrenze=0 bedeutet, dass kein Grenzwert gesetzt ist),
- BisRALter: eventuelles Höchstalter des Versorgungsberechtigten der Dynamisierung der laufenden Invalidenleistung und der Leistungszahlung überhaupt (mit dieser Einstellung stellt eine besondere Form einer abgekürzten Invalidenleistung dar),
- tIR: Schlüssel der für die Angabe der Zahlweise der Invalidenleistung (vgl. dazu unten die Auflistung möglicher Zahlweisen),
- IRFolgejahr: Beschreibt, ab welchem Folgejahr nach Leistungsfall die Zahlung der Invalidenleistung aufgeschoben einsetzen soll (Vorgabewert ist IRFolgeJahr=0: sofort beginnend),
- IRDauer: Parameter zur Angabe der Dauer der Zahlung der Invalidenleistung (0: Rentenzahlung, 1: Kapitalzahlung),
- LaWRNr: Nummer des (Aktiven-)Hinterbliebenenleistungsvektors (in der Regel LaWRNr=3, also Leistungsvektor L3),
- aWRDyn: Dynamiksatzeiner laufenden Hinterbliebenen- bzw. Aktivenhinterbliebenenrente, d. h. Dynamik der entsprechenden laufenden Rente (Dezimalzahl, bspw. 0,02 für eine Dynamik der Rente in Höhe von 2 % bezogen auf den jeweiligen Vorjahreswert),
- aWRWieOft: Parameter, der angibt, alle wie viel Jahre die Dynamisierung der Hinterbliebenenleistung erfolgt (regelmäßig ist aWRWieOft=1 für jährliche Erhöhung der Renten entsprechend des Dynamiksatzes aWRDyn; bspw. würde aWRWieOft=3 mit aWRDyn=0,03 bedeuten, dass die Hinterbliebenenrenten alle drei Jahre um 3 % angehoben werden würde),
- aWRGrenze: Höchst-/Grenzwert der Rentenhöhe für die Dynamisierung der laufenden Hinterbliebenenrente (aWRGrenze=0 bedeutet, dass kein Grenzwert gesetzt ist),

- **BisaWRAlter**: eventuelles Höchstalter des Versorgungsberechtigten der Dynamisierung der laufenden Hinterbliebenenleistung und der Leistungszahlung überhaupt (mit dieser Einstellung stellt eine besondere Form einer abgekürzten Hinterbliebenenleistung dar),
- **taWR**: Schlüssel der für die Angabe der Zahlweise der Hinterbliebenenleistung (vgl. dazu unten die Auflistung möglicher Zahlweisen),
- **aWRFolgejahr**: Beschreibt, ab welchem Folgejahr nach Leistungsfall die Zahlung der Hinterbliebenenleistung aufgeschoben einsetzen soll (Vorgabewert ist $aWRFolgeJahr=0$: sofort beginnend),
- **aWRDauer**: Parameter zur Angabe der Dauer der Zahlung der Hinterbliebenenleistung (0: Rentenzahlung, 1: Kapitalzahlung),
- **LiWRNr**: Nummer des (Invaliden-)Hinterbliebenenleistungsvektors (in der Regel $LiWRNr=4$, also Leistungsvektor L4),
- **iWRDyn**: Dynamiksatzeiner laufenden Invalidenhinterbliebenenrente, d. h. Dynamik der entsprechenden laufenden Rente (Dezimalzahl, bspw. 0,02 für eine Dynamik der Rente in Höhe von 2 % bezogen auf den jeweiligen Vorjahreswert),
- **iWRWieOft**: Parameter, der angibt, alle wie viel Jahre die Dynamisierung der Invalidenhinterbliebenenleistung erfolgt (regelmäßig ist $iWRWieOft=1$ für jährliche Erhöhung der Renten entsprechend des Dynamiksatzes $iWRDyn$; bspw. würde $iWRWieOft=3$ mit $iWRDyn=0,03$ bedeuten, dass die Hinterbliebenenrenten alle drei Jahre um 3 % angehoben werden würde),
- **iWRGrenze**: Höchst-/Grenzwert der Rentenhöhe für die Dynamisierung der laufenden Invalidenhinterbliebenenrente ($iWRGrenze=0$ bedeutet, dass kein Grenzwert gesetzt ist),
- **BisiWRAlter**: eventuelles Höchstalter des Versorgungsberechtigten der Dynamisierung der laufenden Invalidenhinterbliebenenleistung und der Leistungszahlung überhaupt (mit dieser Einstellung stellt eine besondere Form einer abgekürzten Invalidenhinterbliebenenleistung dar),
- **tiWR**: Schlüssel der für die Angabe der Zahlweise der Invalidenhinterbliebenenleistung (vgl. dazu unten die Auflistung möglicher Zahlweisen),
- **iWRFolgejahr**: Beschreibt, ab welchem Folgejahr nach Leistungsfall die Zahlung der Invalidenhinterbliebenenleistung aufgeschoben einsetzen soll (Vorgabewert ist $aWRFolgeJahr=0$: sofort beginnend),
- **iWRDauer**: Parameter zur Angabe der Dauer der Zahlung der Invalidenhinterbliebenenleistung (0: Rentenzahlung, 1: Kapitalzahlung),
- **AlterZus**: Alter des Versorgungsberechtigten am Bilanzstichtag der Zusage bei Entgeltumwandlung (Bilanzjahresende),
- **JahrZus**: Jahreszahl des Bilanzstichtages bei Zusage der Entgeltumwandlung (Bilanzjahresende)
- **AltZusageKey**: Schlüssel für die Angabe der Form der Zusage als Altzusage bzw. Neuzusage (vgl. unten Auflistung der möglichen Schlüsselwert),
- **ZusageForm**: Schlüssel für die Angabe der Form der Zusage als klassische Leistungszusage bzw. als beitragsorientierte Leistungszusage (vgl. unten die Auflistung möglicher ZusageForm-Schlüssel),
- **FinanzForm**: Schlüssel für die Angabe der Finanzierungsform der Zusage in arbeitnehmer- bzw. arbeitgeberfinanzierte Zusage (vgl. unten die Auflistung möglicher FinanzForm-Schlüssel),
- **EUmwAusfinanziert**: Parameter, der den Ausfinanzierungsgrad bei Entgeltumwandlungszusagen angibt (regelmäßig ist $EUmwAusfinanziert=True$ bei echten beitragsorientierten Leistungszusagen),
- **UnvKey**: Schlüssel bzw. Ergebnis der Unverfallbarkeitbestimmung über die dafür bereitgestellte Funktion `Unverfallbar2007(...)`,
- **actUnvAL**: enthält den Wert der ausfinanzierten unverfallbaren Altersleistung zum aktuellen Bilanzstichtag sofern bei beitragsorientierten Versorgungszusagen die Leistungen noch nicht vollständig ausfinanziert sind; vgl. dazu ebenfalls den Parameter `EUmwAusfinanziert`),

- actUnvIL: enthält den Wert der ausfinanzierten unverfallbaren Invalidenleistung zum aktuellen Bilanzstichtag sofern bei beitragsorientierten Versorgungszusagen die Leistungen noch nicht vollständig ausfinanziert sind; vgl. dazu ebenfalls den Parameter EUMwAusfinanziert),
- actUnvAWL: enthält den Wert der ausfinanzierten unverfallbaren (Aktiven-)Hinterbliebenenleistung zum aktuellen Bilanzstichtag sofern bei beitragsorientierten Versorgungszusagen die Leistungen noch nicht vollständig ausfinanziert sind; vgl. dazu ebenfalls den Parameter EUMwAusfinanziert),
- actUnvIWL: enthält den Wert der ausfinanzierten unverfallbaren Invalidenhinterbliebenenleistung zum aktuellen Bilanzstichtag sofern bei beitragsorientierten Versorgungszusagen die Leistungen noch nicht vollständig ausfinanziert sind; vgl. dazu ebenfalls den Parameter EUMwAusfinanziert),
- actUnvZins: enthält Zinssatz für Wertsteigerung des bereits ausfinanzierten Invaliden- bzw. Hinterbliebenenkapitals in den Folgejahren, sofern die entsprechenden Leistung noch nicht vollständig ausfinanziert wurden).

Rückgabewert: Long/Ganzzahl als Adresse der großen Ergebnismatrix-Struktur (PEMatM als Zeiger auf die großer Ergebnismatrix EMatM) im Speicher.

NEU: GetTWmod-Funktion

Syntax: Public Function GetTWmod (ByVal Status As Integer, ByVal GebDat1 As Date, ByVal Alter1 As Integer, ByVal Sex1 As Boolean, ByVal GebDat2 As Date, ByVal Alter2 As Integer, ByVal Sex2 As Boolean, ByVal GesBestand As Boolean, ByVal ALeist As Integer, ByVal ILeist As Integer, ByVal AHLeist As Integer, ByVal IHLeist As Integer, ByVal Jahr As Integer, ByVal TWk As Integer, ByVal Zins As Double, ByVal AblAlter As Integer, ByVal tAR As Long, ByVal LARNr As Integer, ByVal LIRNr As Integer, ByVal ILTyp As Integer, ByVal IRDyn As Double, ByVal IRWieOft As Integer, ByVal IRGrenze As Double, ByVal BisIRAlter As Integer, ByVal tIR As Long, ByVal IRFolgeJahr As Integer, ByVal IRDauer As Integer, ByVal LaWRNr As Integer, ByVal aWRDyn As Double, ByVal aWRWieOft As Integer, ByVal aWRGrenze As Double, ByVal BisawRAlter As Integer, ByVal taWR As Long, ByVal aWRFolgeJahr As Integer, ByVal aWRDauer As Integer, ByVal LiWRNr As Integer, ByVal iWRDyn As Double, ByVal iWRWieOft As Integer, ByVal iWRGrenze As Double, ByVal BisiWRAlter As Integer, ByVal tiWR As Long, ByVal iWRFolgeJahr As Integer, ByVal iWRDauer As Integer, ByVal Alter1Zus As Integer, ByVal JahrZus As Integer, ByVal AltZusagekey As Integer, ByVal ZusageForm As Integer, ByVal FinanzForm As Integer, ByVal EUMwAusfinanziert As Boolean, ByVal UnvKey As Integer, ByVal actUnvAL As Double, ByVal actUnvIL As Double, ByVal actUnvAWL As Double, ByVal actUnvIWL As Double, ByVal actUnvZins As Double) As Long

Beschreibung: Die Funktion GetTWmod wird herangezogen zur Bewertung von Versorgungsverpflichtungen auf Basis des sog. "modifizierten Teilwertverfahrens". Im Unterschied zum klassischen Teilwertverfahren (vgl. Funktionen GetTW2004- bzw. GetTW2005) werden bei der Prämienkalkulation (vgl. sog. "fiktive Jahresnettoprämie) Informationen der bereits erlebten Vergangenheit mit Wahrscheinlichkeit 1 angenommen, was methodisch korrekter zur werten ist. Für die Ermittlung der notwendigen Barwerte der Anwartschaften auf Alters-, Invaliden- sowie Hinterbliebenenleistungen sind als Voraussetzung entsprechend die Leistungsvektoren L1, L2, L3 bzw. L4 geeignet vorzubelegen. Die einzelnen Funktionsparameter haben dabei die folgende Bedeutung:

- Status: Personenstatus des Versorgungsberechtigten (Anwärter, Altersrentner usw.) gemäß Übersicht über die Personenstatus (s. u.),
- GebDat/GebDat1: Geburtsdatum des Versorgungsberechtigten (Anzahl der Tage seit dem 30.12.1899 12.00 Uhr, Nachkommastellen beschreiben Uhrzeit des Tages – irrelevant für reine Datumsangaben),
- Alter/Alter1: Alter des Versorgungsberechtigten bei Statuseintritt, d. h. das Datum des Dienstetrtritts für Versorgungsberechtigte mit Personenstatus „Anwärter“ oder das Datum des Leistungsfalls für Empfänger laufender Leistungen,

- Sex/Sex1: Geschlecht des Versorgungsberechtigten als boolescher Wert (False=weiblich, True=männlich),
- GebDat2/GebDatP: Geburtsdatum des Partners des Versorgungsberechtigten (Anzahl der Tage seit dem 30.12.1899 12.00 Uhr, Nachkommastellen beschreiben Uhrzeit des Tages – irrelevant für reine Datumsangaben),
- Alter2/AlterP: versicherungstechnisches Alter des Partners des Versorgungsberechtigten bei Statuseintritt des Versorgungsberechtigten,
- Sex2/SexP: Geschlecht des Partners des Versorgungsberechtigten (False=weiblich, True=männlich, beachte: gleichgeschlechtliche Beziehungen sind möglich),
- GesBestand: Boolescher Wert zur Angabe, ob die Bewertung im Gesamt- oder im Aktivenbestand erfolgt (True=Gesamtbestand, False=Aktivenbestand),
- ALeist: Angabe zur Berücksichtigung einer Altersleistung (1: Leistungen ab Pensionsalter, 0: keine Leistungen; Hinweis: bei Setzung „0“ werden jegliche Werte im Altersleistungsvektor ignoriert),
- ILeist: Angabe zur Berücksichtigung einer Invalidenleistung (1 : Leistungen bei Invalidität, 0 : keine Leistungen; Hinweis: bei Setzung „0“ werden jegliche Werte im Invalidenleistungsvektor ignoriert),
- AHLeist: Angabe zur Berücksichtigung einer Hinterbliebenenleistung (2: Hinterbliebenenleistungen, 1: Leistungen nur bei Aktiventod, d. h. lediglich Aktiven-Hinterbliebenenleistungen sind vorgegeben, 0: keine Hinterbliebenenleistungen),
- IHLeist: Angabe zur Berücksichtigung einer Invalidenhinterbliebenenleistung (1: Hinterbliebenenleistungen nur bei Invalidentod, 0: keine Leistungen),
- Jahr: Jahreszahl vom Bilanzstichtag des Bilanzjahres des Dienst Eintritts (damit also korrespondierende Jahreszahl des Bilanzstichtages des Beginns des Bilanzjahres; Beispiel: Dienst Eintritt 3.6.1992, dann Jahr=1991, da Bilanzjahresbeginn am 1.1.1992 und damit identisch am 31.12.1991; Angabe ist für die Visualisierung des Äquivalenzprinzips innerhalb des Teilwertverfahrens am Beginn des Wirtschaftsjahres des Dienst Eintritts notwendig),
- TWk: (Zeilen-)Nummer derjenigen Zeile innerhalb der großen Ergebnismatrix, in die das aktuelle Bilanzjahr bzw. genauer der aktuelle Bilanzstichtag fällt; die Ermittlung erfolgt am einfachsten über die dafür bereitgestellte Funktion GetEBilanzjahresbeginnJahreszahlByDate(...); TWk dient zur lediglich zur Identifizierung der sog. Bilanzjahreszeile,
- Zins: für die aktuelle Bewertung relevanter Rechnungszinssatz (Vorgabe für die Steuerbilanz 6 %, Vorgabe für die Handelsbilanz nach den jeweiligen ermittelten Stichtagszinssätzen der Bundesbank gemäß RückAbzinsV),
- AblAlter: Pensionsalter des Versorgungsberechtigten,
- tAR: Schlüssel der für die Angabe der Zahlweise für die Altersleistung (vgl. dazu unten die Auflistung möglicher Zahlweisen),
- LARNr: Nummer Altersleistungsvektors (in der Regel LARNr=1),
- LIRNr: Nummer des Invalidenleistungsvektors (in der Regel LIRNr=2),
- ILTyp: Typ der Invalidenleistung gemäß Erläuterung unten (1: abgekürzte Invalidenleistung bis Pensionsalter, 0: lebenslängliche Invalidenleistung, 2: auf das Pensionsalter aufgeschobene Invalidenleistung, 3: Invalidenleistung mit Sprung im Pensionsalter; beachte bei ILTyp=2 bzw. ILTyp=3 die Sonderbehandlung des Invalidenleistungsvektors),
- IRDyn: Dynamiksatz einer laufenden Invalidenleistung, d. h. Dynamik der Invalidenrente (Dezimalzahl, bspw. 0,02 für eine Dynamik der Rente in Höhe von 2 % bezogen auf den jeweiligen Vorjahreswert),
- IRWieOft: Parameter, der angibt, alle wie viel Jahre die Dynamisierung der Invalidenleistung erfolgt (regelmäßig ist IRWieOft=1 für jährliche Erhöhung der Renten entsprechend des Dynamiksatzes IRDyn; bspw. würde IRWieOft=3 mit IRDyn=0,03 bedeuten, dass die Invalidenrenten alle drei Jahre um 3 % angehoben werden würde),

- IRGrenze: Höchst-/Grenzwert der Rentenhöhe für die Dynamisierung der laufenden Invalidenrente (IRGrenze=0 bedeutet, dass kein Grenzwert gesetzt ist),
- BisIRAlter: eventuelles Höchstalter des Versorgungsberechtigten der Dynamisierung der laufenden Invalidenleistung und der Leistungszahlung überhaupt (mit dieser Einstellung stellt eine besondere Form einer abgekürzten Invalidenleistung dar),
- tIR: Schlüssel der für die Angabe der Zahlweise der Invalidenleistung (vgl. dazu unten die Auflistung möglicher Zahlweisen),
- IRFolgejahr: Beschreibt, ab welchem Folgejahr nach Leistungsfall die Zahlung der Invalidenleistung aufgeschoben einsetzen soll (Vorgabewert ist IRFolgeJahr=0: sofort beginnend),
- IRDauer: Parameter zur Angabe der Dauer der Zahlung der Invalidenleistung (0: Rentenzahlung, 1: Kapitalzahlung),
- LaWRNr: Nummer des (Aktiven-)Hinterbliebenenleistungsvektors (in der Regel LaWRNr=3, also Leistungsvektor L3),
- aWRDyn: Dynamiksatzeiner laufenden Hinterbliebenen- bzw. Aktivenhinterbliebenenrente, d. h. Dynamik der entsprechenden laufenden Rente (Dezimalzahl, bspw. 0,02 für eine Dynamik der Rente in Höhe von 2 % bezogen auf den jeweiligen Vorjahreswert),
- aWRWieOft: Parameter, der angibt, alle wie viel Jahre die Dynamisierung der Hinterbliebenenleistung erfolgt (regelmäßig ist aWRWieOft=1 für jährliche Erhöhung der Renten entsprechend des Dynamiksatzes aWRDyn; bspw. würde aWRWieOft=3 mit aWRDyn=0,03 bedeuten, dass die Hinterbliebenenrenten alle drei Jahre um 3 % angehoben werden würde),
- aWRGrenze: Höchst-/Grenzwert der Rentenhöhe für die Dynamisierung der laufenden Hinterbliebenenrente (aWRGrenze=0 bedeutet, dass kein Grenzwert gesetzt ist),
- BisaWRAlter: eventuelles Höchstalter des Versorgungsberechtigten der Dynamisierung der laufenden Hinterbliebenenleistung und der Leistungszahlung überhaupt (mit dieser Einstellung stellt eine besondere Form einer abgekürzten Hinterbliebenenleistung dar),
- taWR: Schlüssel der für die Angabe der Zahlweise der Hinterbliebenenleistung (vgl. dazu unten die Auflistung möglicher Zahlweisen),
- aWRFolgejahr: Beschreibt, ab welchem Folgejahr nach Leistungsfall die Zahlung der Hinterbliebenenleistung aufgeschoben einsetzen soll (Vorgabewert ist aWRFolgeJahr=0: sofort beginnend),
- aWRDauer: Parameter zur Angabe der Dauer der Zahlung der Hinterbliebenenleistung (0: Rentenzahlung, 1: Kapitalzahlung),
- LiWRNr: Nummer des (Invaliden-)Hinterbliebenenleistungsvektors (in der Regel LiWRNr=4, also Leistungsvektor L4),
- iWRDyn: Dynamiksatzeiner laufenden Invalidenhinterbliebenenrente, d. h. Dynamik der entsprechenden laufenden Rente (Dezimalzahl, bspw. 0,02 für eine Dynamik der Rente in Höhe von 2 % bezogen auf den jeweiligen Vorjahreswert),
- iWRWieOft: Parameter, der angibt, alle wie viel Jahre die Dynamisierung der Invalidenhinterbliebenenleistung erfolgt (regelmäßig ist iWRWieOft=1 für jährliche Erhöhung der Renten entsprechend des Dynamiksatzes iWRDyn; bspw. würde iWRWieOft=3 mit iWRDyn=0,03 bedeuten, dass die Hinterbliebenenrenten alle drei Jahre um 3 % angehoben werden würde),
- iWRGrenze: Höchst-/Grenzwert der Rentenhöhe für die Dynamisierung der laufenden Invalidenhinterbliebenenrente (iWRGrenze=0 bedeutet, dass kein Grenzwert gesetzt ist),
- BisiWRAlter: eventuelles Höchstalter des Versorgungsberechtigten der Dynamisierung der laufenden Invalidenhinterbliebenenleistung und der Leistungszahlung überhaupt (mit dieser Einstellung stellt eine besondere Form einer abgekürzten Invalidenhinterbliebenenleistung dar),
- tiWR: Schlüssel der für die Angabe der Zahlweise der Invalidenhinterbliebenenleistung (vgl. dazu unten die Auflistung möglicher Zahlweisen),

- iWRFolgejahr: Beschreibt, ab welchem Folgejahr nach Leistungsfall die Zahlung der Invalidenhinterbliebenenleistung aufgeschoben einsetzen soll (Vorgabewert ist aWRFolgeJahr=0: sofort beginnend),
- iWRDauer: Parameter zur Angabe der Dauer der Zahlung der Invalidenhinterbliebenenleistung (0: Rentenzahlung, 1: Kapitalzahlung),
- AlterZus: Alter des Versorgungsberechtigten am Bilanzstichtag der Zusage bei Entgeltumwandlung (Bilanzjahresende),
- JahrZus: Jahreszahl des Bilanzstichtages bei Zusage der Entgeltumwandlung (Bilanzjahresende)
- AltZusageKey: Schlüssel für die Angabe der Form der Zusage als Altzusage bzw. Neuzusage (vgl. unten Auflistung der möglichen Schlüsselwert),
- ZusageForm: Schlüssel für die Angabe der Form der Zusage als klassische Leistungszusage bzw. als beitragsorientierte Leistungszusage (vgl. unten die Auflistung möglicher ZusageForm-Schlüssel),
- FinanzForm: Schlüssel für die Angabe der Finanzierungsform der Zusage in arbeitnehmer- bzw. arbeitgeberfinanzierte Zusage (vgl. unten die Auflistung möglicher FinanzForm-Schlüssel),
- EUmwAusfinanziert: Parameter, der den Ausfinanzierungsgrad bei Entgeltumwandlungszusagen angibt (regelmäßig ist EUmwAusfinanziert=True bei echten beitragsorientierten Leistungszusagen),
- UnvKey: Schlüssel bzw. Ergebnis der Unverfallbarkeitbestimmung über die dafür bereitgestellte Funktion Unverfallbar2007(...),
- actUnvAL: enthält den Wert der ausfinanzierten unverfallbaren Altersleistung zum aktuellen Bilanzstichtag sofern bei beitragsorientierten Versorgungszusagen die Leistungen noch nicht vollständig ausfinanziert sind; vgl. dazu ebenfalls den Parameter EUmwAusfinanziert),
- actUnvIL: enthält den Wert der ausfinanzierten unverfallbaren Invalidenleistung zum aktuellen Bilanzstichtag sofern bei beitragsorientierten Versorgungszusagen die Leistungen noch nicht vollständig ausfinanziert sind; vgl. dazu ebenfalls den Parameter EUmwAusfinanziert),
- actUnvAWL: enthält den Wert der ausfinanzierten unverfallbaren (Aktiven-)Hinterbliebenenleistung zum aktuellen Bilanzstichtag sofern bei beitragsorientierten Versorgungszusagen die Leistungen noch nicht vollständig ausfinanziert sind; vgl. dazu ebenfalls den Parameter EUmwAusfinanziert),
- actUnvIWL: enthält den Wert der ausfinanzierten unverfallbaren Invalidenhinterbliebenenleistung zum aktuellen Bilanzstichtag sofern bei beitragsorientierten Versorgungszusagen die Leistungen noch nicht vollständig ausfinanziert sind; vgl. dazu ebenfalls den Parameter EUmwAusfinanziert),
- actUnvZins: enthält Zinssatz für Wertsteigerung des bereits ausfinanzierten Invaliden- bzw. Hinterbliebenenkapitals in den Folgejahren, sofern die entsprechenden Leistung noch nicht vollständig ausfinanziert wurden).

Hinweis: Das modifizierte Teilwertverfahren versteht sich als klassisches Bewertungsverfahren nach „Art der Lebensversicherungsmathematik“, d. h. der Teilwert am Bewertungsstichtag ergibt sich als Deckungskapital aus der Differenz des Barwerts der zukünftigen Leistungen und des Barwerts der zukünftigen Beiträge.

Rückgabewert: Long/Ganzzahl als Adresse der großen Ergebnismatrix-Struktur (PEMatM) im Speicher.

NEU: GetDBO-Funktion

Syntax: Public Function GetDBO (ByVal Status As Integer, ByVal GebDat1 As Date, ByVal Alter1 As Integer, ByVal Sex1 As Boolean, ByVal GebDat2 As Date, ByVal Alter2 As Integer, ByVal Sex2 As Boolean, ByVal GesBestand As Boolean, ByVal ALeist As Integer, ByVal ILeist As Integer, ByVal AHLeist As Integer, ByVal IHLeist As Integer, ByVal Jahr As

Integer, ByVal TWk As Integer, ByVal Zins As Double, ByVal AblAlter As Integer, ByVal tAR As Long, ByVal LARNr As Integer, ByVal LIRNr As Integer, ByVal ILTyp As Integer, ByVal IRDyn As Double, ByVal IRWieOft As Integer, ByVal IRGrenze As Double, ByVal BisIRAlter As Integer, ByVal tIR As Long, ByVal IRFolgeJahr As Integer, ByVal IRDauer As Integer, ByVal LaWRNr As Integer, ByVal aWRDyn As Double, ByVal aWRWieOft As Integer, ByVal aWRGrenze As Double, ByVal BisaWRAlter As Integer, ByVal taWR As Long, ByVal aWRFolgeJahr As Integer, ByVal aWRDauer As Integer, ByVal LiWRNr As Integer, ByVal iWRDyn As Double, ByVal iWRWieOft As Integer, ByVal iWRGrenze As Double, ByVal BisiWRAlter As Integer, ByVal tiWR As Long, ByVal iWRFolgeJahr As Integer, ByVal iWRDauer As Integer, ByVal Alter1Zus As Integer, ByVal JahrZus As Integer, ByVal AltZusagekey As Integer, ByVal ZusageForm As Integer, ByVal FinanzForm As Integer, ByVal EUmwAusfinanziert As Boolean, ByVal UnvKey As Integer, ByVal VerfallDauer As Integer, ByVal Alter1BST As Integer, ByVal FluktAnteil As Double) As Long

Beschreibung: Funktion zur Bewertung einer Versorgungsverpflichtung nach internationalen Bilanzierungsregeln im Sinne der Defined Benefit Obligation (DBO) mit Hilfe der sog. "Projected-Unit-Credit-Methode" (PUC-Methode) auf Basis von Generationentafeln (bspw. Heubeck-Richttafeln RT 2005G). Für die Ermittlung der notwendigen Barwerte der Anwartschaften auf Alters-, Invaliden- sowie Hinterbliebenenleistungen sind als Voraussetzung entsprechend die Leistungsvektoren L1, L2, L3 bzw. L4 geeignet vorzubelegen. Die einzelnen Funktionsparameter haben dabei die folgende Bedeutung:

- Status: Personenstatus des Versorgungsberechtigten (Anwärter, Altersrentner usw.) gemäß Übersicht über die Personenstatus (s. u.),
- GebDat/GebDat1: Geburtsdatum des Versorgungsberechtigten (Anzahl der Tage seit dem 30.12.1899 12.00 Uhr, Nachkommastellen beschreiben Uhrzeit des Tages – irrelevant für reine Datumsangaben),
- Alter/Alter1: Alter des Versorgungsberechtigten bei Statureintritt, d. h. das Datum des Dienstetrtritts für Versorgungsberechtigte mit Personenstatus „Anwärter“ oder das Datum des Leistungsfalls für Empfänger laufender Leistungen,
- Sex/Sex1: Geschlecht des Versorgungsberechtigten als boolescher Wert (False=weiblich, True=männlich),
- GebDat2/GebDatP: Geburtsdatum des Partners des Versorgungsberechtigten (Anzahl der Tage seit dem 30.12.1899 12.00 Uhr, Nachkommastellen beschreiben Uhrzeit des Tages – irrelevant für reine Datumsangaben),
- Alter2/AlterP: versicherungstechnisches Alter des Partners des Versorgungsberechtigten bei Statureintritt des Versorgungsberechtigten,
- Sex2/SexP: Geschlecht des Partners des Versorgungsberechtigten (False=weiblich, True=männlich, beachte: gleichgeschlechtliche Beziehungen sind möglich),
- GesBestand: Boolescher Wert zur Angabe, ob die Bewertung im Gesamt- oder im Aktivenbestand erfolgt (True=Gesamtbestand, False=Aktivenbestand),
- ALeist: Angabe zur Berücksichtigung einer Altersleistung (1: Leistungen ab Pensionsalter, 0: keine Leistungen; Hinweis: bei Setzung „0“ werden jegliche Werte im Altersleistungsvektor ignoriert),
- ILeist: Angabe zur Berücksichtigung einer Invalidenleistung (1 : Leistungen bei Invalidität, 0 : keine Leistungen; Hinweis: bei Setzung „0“ werden jegliche Werte im Invalidenleistungsvektor ignoriert),
- AHLeist: Angabe zur Berücksichtigung einer Hinterbliebenenleistung (2: Hinterbliebenenleistungen, 1: Leistungen nur bei Aktiventod, d. h. lediglich Aktiven-Hinterbliebenenleistungen sind vorgegeben, 0: keine Hinterbliebenenleistungen),
- IHLeist: Angabe zur Berücksichtigung einer Invalidenhinterbliebenenleistung (1: Hinterbliebenenleistungen nur bei Invalidentod, 0: keine Leistungen),
- Jahr: Jahreszahl vom Bilanzstichtag des Bilanzjahres des Dienstetrtritts (damit also korrespondierende Jahreszahl des Bilanzstichtages des Beginns des Bilanzjahres; Beispiel: Dienstetrtritt 3.6.1992, dann Jahr=1991, da Bilanzjahresbeginn am 1.1.1992 und damit identisch am

31.12.1991; Angabe ist für die Visualisierung des Äquivalenzprinzips innerhalb des Teilwertverfahrens am Beginn des Wirtschaftsjahres des Dienst Eintritts notwendig),

- TWk: (Zeilen-)Nummer derjenigen Zeile innerhalb der großen Ergebnismatrix, in die das aktuelle Bilanzjahr bzw. genauer der aktuelle Bilanzstichtag fällt; die Ermittlung erfolgt am einfachsten über die dafür bereitgestellte Funktion GetEABilanzjahrbeginnJahreszahlByDate(...); TWk dient zur lediglich zur Identifizierung der sog. Bilanzjahreszeile,
- Zins: für die aktuelle Bewertung relevanter Rechnungszinssatz (Vorgabe für die Steuerbilanz 6 %, Vorgabe für die Handelsbilanz nach den jeweiligen ermittelten Stichtagszinssätzen der Bundesbank gemäß RückAbzinsV),
- AblAlter: Pensionsalter des Versorgungsberechtigten,
- tAR: Schlüssel der für die Angabe der Zahlweise für die Altersleistung (vgl. dazu unten die Auflistung möglicher Zahlweisen),
- LARNr: Nummer Altersleistungsvektors (in der Regel LARNr=1),
- LIRNr: Nummer des Invalidenleistungsvektors (in der Regel LIRNr=2),
- ILTyp: Typ der Invalidenleistung gemäß Erläuterung unten (1: abgekürzte Invalidenleistung bis Pensionsalter, 0: lebenslängliche Invalidenleistung, 2: auf das Pensionsalter aufgeschobene Invalidenleistung, 3: Invalidenleistung mit Sprung im Pensionsalter; beachte bei ILTyp=2 bzw. ILTyp=3 die Sonderbehandlung des Invalidenleistungsvektors),
- IRDyn: Dynamiksatzeiner laufenden Invalidenleistung, d. h. Dynamik der Invalidenrente (Dezimalzahl, bspw. 0,02 für eine Dynamik der Rente in Höhe von 2 % bezogen auf den jeweiligen Vorjahreswert),
- IRWieOft: Parameter, der angibt, alle wie viel Jahre die Dynamisierung der Invalidenleistung erfolgt (regelmäßig ist IRWieOft=1 für jährliche Erhöhung der Renten entsprechend des Dynamiksatzes IRDyn; bspw. würde IRWieOft=3 mit IRDyn=0,03 bedeuten, dass die Invalidenrenten alle drei Jahre um 3 % angehoben werden würde),
- IRGrenze: Höchst-/Grenzwert der Rentenhöhe für die Dynamisierung der laufenden Invalidenrente (IRGrenze=0 bedeutet, dass kein Grenzwert gesetzt ist),
- BisIRAlter: eventuelles Höchstalter des Versorgungsberechtigten der Dynamisierung der laufenden Invalidenleistung und der Leistungszahlung überhaupt (mit dieser Einstellung stellt eine besondere Form einer abgekürzten Invalidenleistung dar),
- tIR: Schlüssel der für die Angabe der Zahlweise der Invalidenleistung (vgl. dazu unten die Auflistung möglicher Zahlweisen),
- IRFolgejahr: Beschreibt, ab welchem Folgejahr nach Leistungsfall die Zahlung der Invalidenleistung aufgeschoben einsetzen soll (Vorgabewert ist IRFolgeJahr=0: sofort beginnend),
- IRDauer: Parameter zur Angabe der Dauer der Zahlung der Invalidenleistung (0: Rentenzahlung, 1: Kapitalzahlung),
- LaWRNr: Nummer des (Aktiven-)Hinterbliebenenleistungsvektors (in der Regel LaWRNr=3, also Leistungsvektor L3),
- aWRDyn: Dynamiksatzeiner laufenden Hinterbliebenen- bzw. Aktivenhinterbliebenenrente, d. h. Dynamik der entsprechenden laufenden Rente (Dezimalzahl, bspw. 0,02 für eine Dynamik der Rente in Höhe von 2 % bezogen auf den jeweiligen Vorjahreswert),
- aWRWieOft: Parameter, der angibt, alle wie viel Jahre die Dynamisierung der Hinterbliebenenleistung erfolgt (regelmäßig ist aWRWieOft=1 für jährliche Erhöhung der Renten entsprechend des Dynamiksatzes aWRDyn; bspw. würde aWRWieOft=3 mit aWRDyn=0,03 bedeuten, dass die Hinterbliebenenrenten alle drei Jahre um 3 % angehoben werden würde),
- aWRGrenze: Höchst-/Grenzwert der Rentenhöhe für die Dynamisierung der laufenden Hinterbliebenenrente (aWRGrenze=0 bedeutet, dass kein Grenzwert gesetzt ist),

- **BisaWRAlter**: eventuelles Höchstalter des Versorgungsberechtigten der Dynamisierung der laufenden Hinterbliebenenleistung und der Leistungszahlung überhaupt (mit dieser Einstellung stellt eine besondere Form einer abgekürzten Hinterbliebenenleistung dar),
- **taWR**: Schlüssel der für die Angabe der Zahlweise der Hinterbliebenenleistung (vgl. dazu unten die Auflistung möglicher Zahlweisen),
- **aWRFolgejahr**: Beschreibt, ab welchem Folgejahr nach Leistungsfall die Zahlung der Hinterbliebenenleistung aufgeschoben einsetzen soll (Vorgabewert ist $aWRFolgeJahr=0$: sofort beginnend),
- **aWRDauer**: Parameter zur Angabe der Dauer der Zahlung der Hinterbliebenenleistung (0: Rentenzahlung, 1: Kapitalzahlung),
- **LiWRNr**: Nummer des (Invaliden-)Hinterbliebenenleistungsvektors (in der Regel $LiWRNr=4$, also Leistungsvektor L4),
- **iWRDyn**: Dynamiksatzeiner laufenden Invalidenhinterbliebenenrente, d. h. Dynamik der entsprechenden laufenden Rente (Dezimalzahl, bspw. 0,02 für eine Dynamik der Rente in Höhe von 2 % bezogen auf den jeweiligen Vorjahreswert),
- **iWRWieOft**: Parameter, der angibt, alle wie viel Jahre die Dynamisierung der Invalidenhinterbliebenenleistung erfolgt (regelmäßig ist $iWRWieOft=1$ für jährliche Erhöhung der Renten entsprechend des Dynamiksatzes $iWRDyn$; bspw. würde $iWRWieOft=3$ mit $iWRDyn=0,03$ bedeuten, dass die Hinterbliebenenrenten alle drei Jahre um 3 % angehoben werden würde),
- **iWRGrenze**: Höchst-/Grenzwert der Rentenhöhe für die Dynamisierung der laufenden Invalidenhinterbliebenenrente ($iWRGrenze=0$ bedeutet, dass kein Grenzwert gesetzt ist),
- **BisiWRAlter**: eventuelles Höchstalter des Versorgungsberechtigten der Dynamisierung der laufenden Invalidenhinterbliebenenleistung und der Leistungszahlung überhaupt (mit dieser Einstellung stellt eine besondere Form einer abgekürzten Invalidenhinterbliebenenleistung dar),
- **tiWR**: Schlüssel der für die Angabe der Zahlweise der Invalidenhinterbliebenenleistung (vgl. dazu unten die Auflistung möglicher Zahlweisen),
- **iWRFolgejahr**: Beschreibt, ab welchem Folgejahr nach Leistungsfall die Zahlung der Invalidenhinterbliebenenleistung aufgeschoben einsetzen soll (Vorgabewert ist $aWRFolgeJahr=0$: sofort beginnend),
- **iWRDauer**: Parameter zur Angabe der Dauer der Zahlung der Invalidenhinterbliebenenleistung (0: Rentenzahlung, 1: Kapitalzahlung),
- **AlterZus**: Alter des Versorgungsberechtigten am Bilanzstichtag der Zusage bei Entgeltumwandlung (Bilanzjahresende),
- **JahrZus**: Jahreszahl des Bilanzstichtages bei Zusage der Entgeltumwandlung (Bilanzjahresende)
- **AltZusageKey**: Schlüssel für die Angabe der Form der Zusage als Altzusage bzw. Neuzusage (vgl. unten Auflistung der möglichen Schlüsselwert),
- **ZusageForm**: Schlüssel für die Angabe der Form der Zusage als klassische Leistungszusage bzw. als beitragsorientierte Leistungszusage (vgl. unten die Auflistung möglicher ZusageForm-Schlüssel),
- **FinanzForm**: Schlüssel für die Angabe der Finanzierungsform der Zusage in arbeitnehmer- bzw. arbeitgeberfinanzierte Zusage (vgl. unten die Auflistung möglicher FinanzForm-Schlüssel),
- **EUmwAusfinanziert**: Parameter, der den Ausfinanzierungsgrad bei Entgeltumwandlungszusagen angibt (regelmäßig ist $EUmwAusfinanziert=True$ bei echten beitragsorientierten Leistungszusagen),
- **UnvKey**: Schlüssel bzw. Ergebnis der Unverfallbarkeitbestimmung über die dafür bereitgestellte Funktion `Unverfallbar2007(...)`,
- **VerfallDauer**: Anzahl der Jahre mit verfallbarer Anwartschaft ab Zusagealter als Ergebnis des Aufrufs der dazu bereitgestellten Funktion `Unverfallbar2007(...)` durch Auslesen des variablen Parameters `AnzVerfJahre`, vgl. Funktion `Unverfallbar2007` weiter unten),

- AlterBST: Alter des Versorgungsberechtigten am aktuellen Bilanzstichtag,
- FluktAnteil: Parameter zur Angabe des Prozentsatzes der angenommenen Fluktuation zur Berücksichtigung bei der Barwertermittlung.

Hinweis: Die PUC-Methode stellt ein Anwartschaftsbarwertverfahren dar, bei dem zur Bewertung der Verpflichtungen zum Bewertungsstichtag die jeweiligen Leistungen zeiträtierlich mit deren aus Sicht des Bewertungsstichtages erdientem Anteil gewichtet werden. Die Funktion weist in der zu berechnenden Ergebnismatrix neben der Defined Benefit Obligation (DBO) als Anwartschaftsbarwert zum aktuellen Bilanzstichtag ebenfalls die erwartete Defined Benefit Obligation (EDBO) als Projektion auf den nächsten Bilanzstichtag aus. Die entsprechenden Spalten erhalten als Spaltenbezeichnung die Werte DBO bzw. EDBO.

Rückgabewert: Long/Ganzzahl als Adresse der großen Ergebnismatrix-Struktur (PEMatM) im Speicher.

NEU: GetPUC-Funktion

Syntax: Public Function GetPUC (ByVal Status As Integer, ByVal GebDat1 As Date, ByVal Alter1 As Integer, ByVal Sex1 As Boolean, ByVal GebDat2 As Date, ByVal Alter2 As Integer, ByVal Sex2 As Boolean, ByVal GesBestand As Boolean, ByVal ALeist As Integer, ByVal ILeist As Integer, ByVal AHLeist As Integer, ByVal IHLeist As Integer, ByVal Jahr As Integer, ByVal TWk As Integer, ByVal Zins As Double, ByVal AblAlter As Integer, ByVal tAR As Long, ByVal LARNr As Integer, ByVal LIRNr As Integer, ByVal ILTyp As Integer, ByVal IRDyn As Double, ByVal IRWieOft As Integer, ByVal IRGrenze As Double, ByVal BisIRAlter As Integer, ByVal tIR As Long, ByVal IRFolgeJahr As Integer, ByVal IRDauer As Integer, ByVal LaWRNr As Integer, ByVal aWRDyn As Double, ByVal aWRWieOft As Integer, ByVal aWRGrenze As Double, ByVal BisawRAlter As Integer, ByVal taWR As Long, ByVal aWRFolgejahr As Integer, ByVal aWRDauer As Integer, ByVal LiWRNr As Integer, ByVal iWRDyn As Double, ByVal iWRWieOft As Integer, ByVal iWRGrenze As Double, ByVal BisiWRAlter As Integer, ByVal tiWR As Long, ByVal iWRFolgejahr As Integer, ByVal iWRDauer As Integer, ByVal Alter1Zus As Integer, ByVal JahrZus As Integer, ByVal AltZusagekey As Integer, ByVal ZusageForm As Integer, ByVal FinanzForm As Integer, ByVal EUmwAusfinanziert As Boolean, ByVal UnvKey As Integer, ByVal VerfallDauer As Integer, ByVal Alter1BST As Integer, ByVal FluktAnteil As Double) As Long

Beschreibung: Funktion zur Bewertung einer Versorgungsverpflichtung mit Hilfe der sog. "Projected-Unit-Credit-Methode" (PUC-Methode) auf Basis von Generationentafeln (bspw. Heubeck-Richttafeln RT 2005G). Die PUC-Methode stellt dabei ein gültiges Bewertungsverfahren innerhalb der deutschen Handelsbilanz (vgl. Bilanzrechtsmodernisierungsgesetz/BilMoG) dar. Für die Ermittlung der notwendigen Barwerte der Anwartschaften auf Alters-, Invaliden- sowie Hinterbliebenenleistungen sind als Voraussetzung entsprechend die Leistungsvektoren L1, L2, L3 bzw. L4 geeignet vorzubelegen. Die einzelnen Funktionsparameter haben dabei die folgende Bedeutung:

- Status: Personenstatus des Versorgungsberechtigten (Anwärter, Altersrentner usw.) gemäß Übersicht über die Personenstatus (s. u.),
- GebDat/GebDat1: Geburtsdatum des Versorgungsberechtigten (Anzahl der Tage seit dem 30.12.1899 12.00 Uhr, Nachkommastellen beschreiben Uhrzeit des Tages – irrelevant für reine Datumsangaben),
- Alter/Alter1: Alter des Versorgungsberechtigten bei Statuseintritt, d. h. das Datum des Dienstetrtritts für Versorgungsberechtigte mit Personenstatus „Anwärter“ oder das Datum des Leistungsfalls für Empfänger laufender Leistungen,
- Sex/Sex1: Geschlecht des Versorgungsberechtigten als boolescher Wert (False=weiblich, True=männlich),
- GebDat2/GebDatP: Geburtsdatum des Partners des Versorgungsberechtigten (Anzahl der Tage seit dem 30.12.1899 12.00 Uhr, Nachkommastellen beschreiben Uhrzeit des Tages – irrelevant für reine Datumsangaben),

- Alter2/AlterP: versicherungstechnisches Alter des Partners des Versorgungsberechtigten bei Statuseintritt des Versorgungsberechtigten,
- Sex2/SexP: Geschlecht des Partners des Versorgungsberechtigten (False=weiblich, True=männlich, beachte: gleichgeschlechtliche Beziehungen sind möglich),
- GesBestand: Boolescher Wert zur Angabe, ob die Bewertung im Gesamt- oder im Aktivenbestand erfolgt (True=Gesamtbestand, False=Aktivenbestand),
- ALeist: Angabe zur Berücksichtigung einer Altersleistung (1: Leistungen ab Pensionsalter, 0: keine Leistungen; Hinweis: bei Setzung „0“ werden jegliche Werte im Altersleistungsvektor ignoriert),
- lLeist: Angabe zur Berücksichtigung einer Invalidenleistung (1 : Leistungen bei Invalidität, 0 : keine Leistungen; Hinweis: bei Setzung „0“ werden jegliche Werte im Invalidenleistungsvektor ignoriert),
- AHLeist: Angabe zur Berücksichtigung einer Hinterbliebenenleistung (2: Hinterbliebenenleistungen, 1: Leistungen nur bei Aktiventod, d. h. lediglich Aktiven-Hinterbliebenenleistungen sind vorgegeben, 0: keine Hinterbliebenenleistungen),
- lHLeist: Angabe zur Berücksichtigung einer Invalidenhinterbliebenenleistung (1: Hinterbliebenenleistungen nur bei Invalidentod, 0: keine Leistungen),
- Jahr: Jahreszahl vom Bilanzstichtag des Bilanzjahres des Dienst Eintritts (damit also korrespondierende Jahreszahl des Bilanzstichtages des Beginns des Bilanzjahres; Beispiel: Dienst Eintritt 3.6.1992, dann Jahr=1991, da Bilanzjahresbeginn am 1.1.1992 und damit identisch am 31.12.1991; Angabe ist für die Visualisierung des Äquivalenzprinzips innerhalb des Teilwertverfahrens am Beginn des Wirtschaftsjahres des Dienst Eintritts notwendig),
- TWk: (Zeilen-)Nummer derjenigen Zeile innerhalb der großen Ergebnismatrix, in die das aktuelle Bilanzjahr bzw. genauer der aktuelle Bilanzstichtag fällt; die Ermittlung erfolgt am einfachsten über die dafür bereitgestellte Funktion GetEABilanzjahrbeginnJahreszahlByDate(...); TWk dient zur lediglich zur Identifizierung der sog. Bilanzjahreszeile,
- Zins: für die aktuelle Bewertung relevanter Rechnungszinssatz (Vorgabe für die Steuerbilanz 6 %, Vorgabe für die Handelsbilanz nach den jeweiligen ermittelten Stichtagszinssätzen der Bundesbank gemäß RückAbzinsV),
- AblAlter: Pensionsalter des Versorgungsberechtigten,
- tAR: Schlüssel der für die Angabe der Zahlweise für die Altersleistung (vgl. dazu unten die Auflistung möglicher Zahlweisen),
- LARNr: Nummer Altersleistungsvektors (in der Regel LARNr=1),
- LIRNr: Nummer des Invalidenleistungsvektors (in der Regel LIRNr=2),
- ILTyp: Typ der Invalidenleistung gemäß Erläuterung unten (1: abgekürzte Invalidenleistung bis Pensionsalter, 0: lebenslängliche Invalidenleistung, 2: auf das Pensionsalter aufgeschobene Invalidenleistung, 3: Invalidenleistung mit Sprung im Pensionsalter; beachte bei ILTyp=2 bzw. ILTyp=3 die Sonderbehandlung des Invalidenleistungsvektors),
- IRDyn: Dynamiksatzeiner laufenden Invalidenleistung, d. h. Dynamik der Invalidenrente (Dezimalzahl, bspw. 0,02 für eine Dynamik der Rente in Höhe von 2 % bezogen auf den jeweiligen Vorjahreswert),
- IRWieOft: Parameter, der angibt, alle wie viel Jahre die Dynamisierung der Invalidenleistung erfolgt (regelmäßig ist IRWieOft=1 für jährliche Erhöhung der Renten entsprechend des Dynamiksatzes IRDyn; bspw. würde IRWieOft=3 mit IRDyn=0,03 bedeuten, dass die Invalidenrenten alle drei Jahre um 3 % angehoben werden würde),
- IRGrenze: Höchst-/Grenzwert der Rentenhöhe für die Dynamisierung der laufenden Invalidenrente (IRGrenze=0 bedeutet, dass kein Grenzwert gesetzt ist),
- BisIRAlter: eventuelles Höchstalter des Versorgungsberechtigten der Dynamisierung der laufenden Invalidenleistung und der Leistungszahlung überhaupt (mit dieser Einstellung stellt eine besondere Form einer abgekürzten Invalidenleistung dar),

- tIR: Schlüssel der für die Angabe der Zahlweise der Invalidenleistung (vgl. dazu unten die Auflistung möglicher Zahlweisen),
- IRFolgejahr: Beschreibt, ab welchem Folgejahr nach Leistungsfall die Zahlung der Invalidenleistung aufgeschoben einsetzen soll (Vorgabewert ist IRFolgeJahr=0: sofort beginnend),
- IRDauer: Parameter zur Angabe der Dauer der Zahlung der Invalidenleistung (0: Rentenzahlung, 1: Kapitalzahlung),
- LaWRNr: Nummer des (Aktiven-)Hinterbliebenenleistungsvektors (in der Regel LaWRNr=3, also Leistungsvektor L3),
- aWRDyn: Dynamiksatzeiner laufenden Hinterbliebenen- bzw. Aktivenhinterbliebenenrente, d. h. Dynamik der entsprechenden laufenden Rente (Dezimalzahl, bspw. 0,02 für eine Dynamik der Rente in Höhe von 2 % bezogen auf den jeweiligen Vorjahreswert),
- aWRWieOft: Parameter, der angibt, alle wie viel Jahre die Dynamisierung der Hinterbliebenenleistung erfolgt (regelmäßig ist aWRWieOft=1 für jährliche Erhöhung der Renten entsprechend des Dynamiksatzes aWRDyn; bspw. würde aWRWieOft=3 mit aWRDyn=0,03 bedeuten, dass die Hinterbliebenenrenten alle drei Jahre um 3 % angehoben werden würde),
- aWRGrenze: Höchst-/Grenzwert der Rentenhöhe für die Dynamisierung der laufenden Hinterbliebenenrente (aWRGrenze=0 bedeutet, dass kein Grenzwert gesetzt ist),
- BisaWRAlter: eventuelles Höchstalter des Versorgungsberechtigten der Dynamisierung der laufenden Hinterbliebenenleistung und der Leistungszahlung überhaupt (mit dieser Einstellung stellt eine besondere Form einer abgekürzten Hinterbliebenenleistung dar),
- taWR: Schlüssel der für die Angabe der Zahlweise der Hinterbliebenenleistung (vgl. dazu unten die Auflistung möglicher Zahlweisen),
- aWRFolgejahr: Beschreibt, ab welchem Folgejahr nach Leistungsfall die Zahlung der Hinterbliebenenleistung aufgeschoben einsetzen soll (Vorgabewert ist aWRFolgeJahr=0: sofort beginnend),
- aWRDauer: Parameter zur Angabe der Dauer der Zahlung der Hinterbliebenenleistung (0: Rentenzahlung, 1: Kapitalzahlung),
- LiWRNr: Nummer des (Invaliden-)Hinterbliebenenleistungsvektors (in der Regel LiWRNr=4, also Leistungsvektor L4),
- iWRDyn: Dynamiksatzeiner laufenden Invalidenhinterbliebenenrente, d. h. Dynamik der entsprechenden laufenden Rente (Dezimalzahl, bspw. 0,02 für eine Dynamik der Rente in Höhe von 2 % bezogen auf den jeweiligen Vorjahreswert),
- iWRWieOft: Parameter, der angibt, alle wie viel Jahre die Dynamisierung der Invalidenhinterbliebenenleistung erfolgt (regelmäßig ist iWRWieOft=1 für jährliche Erhöhung der Renten entsprechend des Dynamiksatzes iWRDyn; bspw. würde iWRWieOft=3 mit iWRDyn=0,03 bedeuten, dass die Hinterbliebenenrenten alle drei Jahre um 3 % angehoben werden würde),
- iWRGrenze: Höchst-/Grenzwert der Rentenhöhe für die Dynamisierung der laufenden Invalidenhinterbliebenenrente (iWRGrenze=0 bedeutet, dass kein Grenzwert gesetzt ist),
- BisiWRAlter: eventuelles Höchstalter des Versorgungsberechtigten der Dynamisierung der laufenden Invalidenhinterbliebenenleistung und der Leistungszahlung überhaupt (mit dieser Einstellung stellt eine besondere Form einer abgekürzten Invalidenhinterbliebenenleistung dar),
- tiWR: Schlüssel der für die Angabe der Zahlweise der Invalidenhinterbliebenenleistung (vgl. dazu unten die Auflistung möglicher Zahlweisen),
- iWRFolgejahr: Beschreibt, ab welchem Folgejahr nach Leistungsfall die Zahlung der Invalidenhinterbliebenenleistung aufgeschoben einsetzen soll (Vorgabewert ist iWRFolgeJahr=0: sofort beginnend),
- iWRDauer: Parameter zur Angabe der Dauer der Zahlung der Invalidenhinterbliebenenleistung (0: Rentenzahlung, 1: Kapitalzahlung),

- AlterZus: Alter des Versorgungsberechtigten am Bilanzstichtag der Zusage bei Entgeltumwandlung (Bilanzjahresende),
- JahrZus: Jahreszahl des Bilanzstichtages bei Zusage der Entgeltumwandlung (Bilanzjahresende)
- AltZusageKey: Schlüssel für die Angabe der Form der Zusage als Altzusage bzw. Neuzusage (vgl. unten Auflistung der möglichen Schlüsselwert),
- ZusageForm: Schlüssel für die Angabe der Form der Zusage als klassische Leistungszusage bzw. als beitragsorientierte Leistungszusage (vgl. unten die Auflistung möglicher ZusageForm-Schlüssel),
- FinanzForm: Schlüssel für die Angabe der Finanzierungsform der Zusage in arbeitnehmer- bzw. arbeitgeberfinanzierte Zusage (vgl. unten die Auflistung möglicher FinanzForm-Schlüssel),
- EUmwAusfinanziert: Parameter, der den Ausfinanzierungsgrad bei Entgeltumwandlungszusagen angibt (regelmäßig ist EUmwAusfinanziert=True bei echten beitragsorientierten Leistungszusagen),
- UnvKey: Schlüssel bzw. Ergebnis der Unverfallbarkeitbestimmung über die dafür bereitgestellte Funktion Unverfallbar2007(...),
- VerfallDauer: Anzahl der Jahre mit verfallbarer Anwartschaft ab Zusagealter als Ergebnis des Aufrufs der dazu bereitgestellten Funktion Unverfallbar2007(...) durch Auslesen des variablen Parameters AnzVerfJahre, vgl. Funktion Unverfallbar2007 weiter unten),
- AlterBST: Alter des Versorgungsberechtigten am aktuellen Bilanzstichtag,
- FluktAnteil: Parameter zur Angabe des Prozentsatzes der angenommenen Fluktuation zur Berücksichtigung bei der Barwertermittlung.

Hinweis: Die PUC-Methode stellt ein Anwartschaftsbarwertverfahren dar, bei dem zur Bewertung der Verpflichtungen zum Bewertungsstichtag die jeweiligen Leistungen zeiträtierlich mit deren aus Sicht des Bewertungsstichtages erdientem Anteil gewichtet werden. Der Berechnungsaufwurf für die GetPUC-Methode wurde abgeleitet von GetDBO-Methode. Als Unterschied wird bei der PUC-Methode keine Berechnung der erwarteten Defined Benefit Obligation (EDBO) als Projektion auf den nächsten Bilanzstichtag vorgenommen. Die Ergebnisspalte zur Angabe des Anwartschaftsbarwertes bekommt die Überschrift „AnwXXX“, wobei „XXX“ der ursprüngliche Gesamtbarwertbezeichner bezeichnet.

Rückgabewert: Long/Ganzzahl als Adresse der großen Ergebnismatrix-Struktur (PEMatM) im Speicher.

NEU: Unverfallbar2007-Funktion

Syntax: Public Function Unverfallbar2007 (ByVal GebDatum As Date, ByVal DEDatum As Date, ByVal ZusDatum As Date, ByVal StichtagDatum As Date, ByVal AnzVerfallJahre As Integer) As Integer

Beschreibung: Ermittelt den Unverfallbarkeitsstatus für die Feststellung der gesetzlichen Unverfallbarkeit. Die ersten drei Datumsangaben werden in aufsteigender Reihenfolge erwartet, ansonsten entsprechend sortiert interpretiert.

Hinweis: Der Referenzparameter AnzVerfJahre gibt die indirekt die Anzahl der Jahre mit verfallbarer Anwartschaft ab Zusagealter zurück.

Rückgabewert: Integer/Ganzzahl als Schlüssel zur Beschreibung des Unverfallbarkeitsstatus. Mögliche Werte sind:

- 1: vierter Datumswert „Stichdatum“ ist nicht das jüngste Datum,
- 0: keine Unverfallbarkeit,
- 1: unverfallbare Altzusage vor 1.1.2001 (Alter >=35J, >=10 Zusagejahre),
- 2: unverfallbare Altzusage (Alter >=35J, >=12 Dienstjahre, >=3 Zusagejahre),
- 3: unverfallbare Altzusage mit Übergangsregelung (Alter >= 30 Jahre, Zusage vor 1.1.2001,

- StichDatum >= 31.12.2005),
- 4: unverfallbare Mittel-Neuzusage ab 1.1.2001 und bis 31.12.2008 (Alter >= 30 Jahre, >=5 Zusagejahre),
 - 5: unverfallbare Neuzusage ab 1.1.2009 (Alter >= 25 Jahre, >=5 Zusagejahre),
 - 6: unverfallbare Mittel-Neuzusage nach Übergangsregelung vor 1.1.2009 (Alter >= 25 Jahre, StichDatum >= 31.12.2013).

Funktionen zum Auslesen der Ergebnismatrix

GetHeaderMatrixPointer-Funktion

Syntax: `Public Function GetHeaderMatrixPointer() As Long`

Beschreibung: Liefert den Zeiger auf die HeaderMatrix-Struktur (PEMatSpez). Vorausgegangen sein muss dem Aufruf von GetHeaderMatrixPointer ein Aufruf einer Methode zur Erzeugung der Ergebnismatrix, wie zum Beispiel GetTW2005(...), so dass in jeden Fall eine gültige Berechnung stattgefunden hat. In der HeaderMatrix-Struktur sind neben der gesamten Überschriftenzeile der Ergebnismatrix zudem Informationen über die je Spalte verwendete Gesamtanzahl von Zeichen inkl. Tausendertrennzeichen und Dezimaltrennzeichen und darüber hinaus die Anzahl verwendeter Nachkommastellen.

Hinweis: Der Zugriff auf die einzelnen Elemente der HeaderMatrix-Struktur wird durch das PMAktuar-System nicht weiter unterstützt. Für die wesentlichen Informationen zu den einzelnen Spaltenüberschriften sei auf die Funktionen und Prozeduren GetColumnTitleArray, GetColumnTitles, GetColumnTitle, GetColumnNumber und PutHeaderToWorksheet verwiesen.

Rückgabewert: Long/Ganzzahl als Adresse der HeaderMatrix-Struktur im Speicher.

GetErgebnisMatrixPointer-Funktion

Syntax: `Public Function GetErgebnisMatrixPointer() As Long`

Beschreibung: Liefert den Zeiger auf die Ergebnismatrix-Struktur (PEMatM). Vorausgegangen sein muss dem Aufruf von GetErgebnisMatrixPointer ein Aufruf einer Methode zur Erzeugung der Ergebnismatrix, wie zum Beispiel GetTW2005(...), so dass in jedem Fall eine gültige Berechnung stattgefunden hat. Die Ergebnismatrix-Struktur ist ein besonderes auf Microsoft Excel abgestimmtes Konstrukt (XLArray) zur Aufnahme einer zweidimensionalen Matrix aus Gleitkommawerten (Double), die neben zwei Angaben (jeweils 2-Byte-Word) für die genaue Anzahl der Zeilen und der Spalten der Matrix, ebenfalls die Ergebnismatrix bestehend aus Gleitkommawerten (Double) enthält.

Hinweis: Der direkte Zugriff auf die Ergebnismatrix-Struktur sei nicht empfohlen. Benutzen Sie zum Auslesen der Informationen der Ergebnismatrixstruktur bitte die umfassenden Funktionen und Prozeduren GetRowCount, GetColCount, GetMatrixElement, GetMatrixAsArray, GetMatrixColumn, GetMatrixColumnPart, GetMatrixColumnPartByAge, GetMatrixColumnByTitle, PutMatrixToWorksheet, PutMatrixColumnToWorksheet, PutMatrixColumnToWorksheetByRow und PutMatrixColumnToWorksheetByAge.

Rückgabewert: Long/Ganzzahl als Adresse der Ergebnismatrix-Struktur im Speicher.

GetRowCount-Funktion

Syntax: `Public Function GetRowCount(ByVal pMatrix As Long) As Integer`

Beschreibung: (Beschreibung folgt...)

Rückgabewert: Integer

GetColCount-Funktion

Syntax: `Public Function GetColCount(ByVal pMatrix As Long) As Integer`

Beschreibung: (Beschreibung folgt...)

Rückgabewert: Integer

GetColumnTitleArray-Funktion

Syntax: `Public Function GetColumnTitleArray() As Variant`

Beschreibung: Die Funktion liefert die Überschriftenzeile der HeaderMatrix-Struktur als VBA-Array zurück. Dem Aufruf dieser Funktion muss der Aufruf einer Funktion vorausgegangen sein, die die große Ergebnismatrix fehlerfrei füllt. Ein gültiger Zeiger auf die große Ergebnismatrix muss existieren.

Rückgabewert: Variant/VBA-Array mit Einträgen vom Typ String, das je Eintrag eine Spaltenüberschrift der HeaderStruktur-Matrix abbildet und somit die Überschriftenzeile der großen Ergebnismatrix darstellt. Da das Array einen Zeilenvektor darstellt, muss für die Ausgabe in einem Tabellenblatt nicht die Excel-Funktion `Application.WorksheetFunction.Transpose(...)` verwendet werden.

GetColumnTitles-Funktion

Syntax: `Public Function GetColumnTitles() As String`

Beschreibung: Diese Funktion liefert die Überschriftenzeile der HeaderMatrix-Struktur als Zeichenkette wieder, die die einzelnen Überschriften für die große Ergebnismatrix durch Semikolons getrennt enthält.

Rückgabewert: String/Zeichenkette als Semikolon-getrennte Liste der Überschriften der HeaderMatrix-Struktur, die die Überschriften der großen Ergebnismatrix wiedergibt.

GetColumnTitle-Funktion

Syntax: `Public Function GetColumnTitle(ByVal ColNr As Integer) As String`

Beschreibung: Diese Funktion liefert für die durch ColNr (Integer) bezeichnete Spalte die Überschrift als Zeichenkette wieder.

Rückgabewert: String/Zeichenkette mit der Überschrift der durch den Parameter ColNr angegebenen Spalte für die große Ergebnismatrix.

GetColumnNumber-Funktion

Syntax: `Public Function GetColumnNumber(ByVal ColTitle As String, Optional MatchExactly As Boolean = False) As Integer`

Beschreibung: Die Funktion bestimmt die Nummer der durch ColTitle (String) angegebenen Spalte beginnend ab 1 innerhalb der Ergebnismatrix. Der optionale Parameter MatchExactly = True gibt dabei an, dass für die Vergleiche völlige Übereinstimmung gelten muss, wohingegen für MatchExactly = False nur die Übereinstimmung so vieler Zeichen relevant ist, wie der Länge von ColTitle entspricht. Für MatchExactly = False ist also zum Beispiel bei dem Suchkriterium "EA" als erlaubtes Ergebnis die Spaltenbezeichnung "EAX" oder "EAY" neben „EA“ ebenfalls möglich. Das erste solche Vorkommen wird festgehalten und dessen Spaltenindex zurückgeliefert.

Rückgabewert: Integer/Ganzzahl, die den Spaltenindex der gefundenen Spalte wiedergibt; wird keine passende Übereinstimmung gefunden, so wird als Rückgabewert -1 geliefert.

GetMatrixAsArray-Funktion

Syntax: `Public Function GetMatrixAsArray() As Variant`

Beschreibung: Liefert die Ergebnismatrix (die große Ergebnismatrix) als zweidimensionales VBA-Array zurück. Vorausgegangen muss dem Aufruf von GetMatrixAsArray ein Aufruf einer Funktion, die die große Ergebnismatrix füllt.

Hinweis: Der Rückgabewert stellt ein zweidimensionales VBA-Array dar. Besonderheit ist hierbei der Zugriff auf die Matrix in Col-Major-Order, d. h. es werden zuerst die Spalten adressiert, dann die Zeilen. Zur richtigen spaltenweisen Darstellung des zweidimensionalen Arrays in einem Tabellenblatt verwenden Sie die Excel-Funktion `Application.WorksheetFunction.Transpose(...)`.

Rückgabewert: Variant/Enthält ein zweidimensionales VBA-Array mit Gleitkommazahlen (Double) als Einträgen in Spalten-Zeilen-Form.

GetMatrixElement-Funktion

Syntax: `Public Function GetMatrixElement(ByVal Zeile As Integer, ByVal Spalte As Integer) As Double`

Beschreibung: Diese Funktion liefert nach erfolgter Berechnung der Ergebnismatrix das durch Parameter Zeile und Spalte angegebene Element der Matrix als Gleitkommazahl.

Rückgabewert: Double/Gleitkommazahl mit Eintrag `Matrix(Zeile, Spalte)`.

GetMatrixColumn-Funktion

Syntax: `Public Function GetMatrixColumn(ByVal ColNr As Integer) As Variant`

Beschreibung: Liefert die durch ColNr (Integer) angegebene Spalte der Ergebnismatrix zurück. Dem Aufruf dieser Funktion muss der Aufruf einer Funktion vorausgegangen sein, die die große Ergebnismatrix füllt.

Hinweis: Beachten Sie, dass das Rückgabe-Array ein Zeilenvektor ist. Zur Darstellung als Spalte innerhalb eines Tabellenblatt ist aus diesem Grund die Excel-Funktion Application.WorksheetFunction.Transpose(...) zu verwenden. Die Anzahl der Elemente des Arrays bestimmen Sie mit den Array-Funktionen LBound(...) und UBound(...).

Rückgabewert: Variant/Enthält den eindimensionalen Array mit Gleitkommazahlen (Double) als Einträgen in Form eines Zeilenvektors.

GetMatrixColumnPart-Funktion

Syntax: `Public Function GetMatrixColumnPart(ByVal ColNr As Integer, ByVal StartFromRow As Integer, Optional ByVal EndAtRow As Integer = 0) As Variant`

Beschreibung: Liefert die durch ColNr (Integer) angegebene Spalte der Ergebnismatrix zurück. Dabei werden nur die Einträge der Zeilen StartFromRow bis EndAtRow zurückgegeben. Dem Aufruf dieser Funktion muss der Aufruf einer Funktion vorausgegangen sein, die die große Ergebnismatrix füllt. Der optionale Parameter EndAtRow kann ignoriert werden, wenn alle Werte der Spalte bis zum Tafelende zurückgeliefert werden sollen, was identisch ist mit der Setzung des Parameters EndAtRow = 0.

Hinweis: Beachten Sie, dass das Rückgabe-Array ein Zeilenvektor ist. Zur Darstellung als Spalte innerhalb eines Tabellenblatt ist aus diesem Grund die Excel-Funktion Application.WorksheetFunction.Transpose(...) zu verwenden. Die Anzahl der Elemente des Arrays bestimmen Sie mit den Array-Funktionen LBound(...) und UBound(...).

Rückgabewert: Variant/Enthält den eindimensionalen Array mit Gleitkommazahlen (Double) als Einträgen in Form eines Zeilenvektors.

GetMatrixColumnPartByAge-Funktion

Syntax: `Public Function GetMatrixColumnPartByAge(ByVal ColNr As Integer, ByVal StartFromAge As Integer, Optional ByVal EndAtAge As Integer = 0) As Variant`

Beschreibung: Liefert die durch ColNr (Integer) angegebene Spalte der Ergebnismatrix zurück. Dabei werden nur die Einträge der Zeilen mit dem Alter StartFromAge (Integer) bis EndAtAge (Integer) zurückgegeben. Dem Aufruf dieser Funktion muss der Aufruf einer Funktion vorausgegangen sein, die die große Ergebnismatrix füllt. Der optionale Parameter EndAtAge kann ignoriert werden, wenn alle Werte der Spalte bis zum Tafelende zurückgeliefert werden sollen, was identisch ist mit der Setzung des Parameters EndAtAge = 0.

Hinweis: Beachten Sie, dass das Rückgabe-Array ein Zeilenvektor ist. Zur Darstellung als Spalte innerhalb eines Tabellenblatt ist aus diesem Grund die Excel-Funktion Application.WorksheetFunction.Transpose(...) zu verwenden. Die Anzahl der Elemente des Arrays bestimmen Sie mit den Array-Funktionen LBound(...) und UBound(...).

Rückgabewert: Variant/Enthält den eindimensionalen Array mit Gleitkommazahlen (Double) als Einträgen in Form eines Zeilenvektors.

GetMatrixColumnByTitle-Funktion

Syntax: `Public Function GetMatrixColumnByTitle(ByVal ColTitle As String) As Variant`

Beschreibung: Liefert die durch ColTitle (ColTitle) angegebene Spalte der Ergebnismatrix zurück. Dem Aufruf dieser Funktion muss der Aufruf einer Funktion vorausgegangen sein, die die große Ergebnismatrix füllt. Der optionale Parameter EndAtAge kann ignoriert werden, wenn alle Werte der Spalte bis zum Tafelende zurückgeliefert werden sollen, was identisch ist mit der Setzung des Parameters EndAtAge = 0.

Hinweis: Beachten Sie, dass das Rückgabe-Array ein Zeilenvektor ist. Zur Darstellung als Spalte innerhalb eines Tabellenblatt ist aus diesem Grund die Excel-Funktion Application.WorksheetFunction.Transpose(...) zu verwenden. Die Anzahl der Elemente des Arrays bestimmen Sie mit den Array-Funktionen LBound(...) und UBound(...).

Rückgabewert: Variant/Enthält den eindimensionalen Array mit Gleitkommazahlen (Double) als Einträgen in Form eines Zeilenvektors.

Prozeduren zur Ausgabe in Tabellenblätter

PutMatrixToWorksheet-Prozedur

Syntax: `Public Sub PutMatrixToWorksheet(ByVal XLArrayPtr As Long, TopLeftCell As Range)`

Beschreibung: Diese Prozedur überträgt die durch den Zeiger XLArrayPtr (Long) angegebene Ergebnismatrix aus PMAktuar in ein Excel-Tabellenblatt. Durch den Parameter TopLeftCell (Range) wird die obere, linke Zelle innerhalb des Tabellenblattes festgelegt.

PutHeaderToWorksheet-Prozedur

Syntax: `Public Sub PutHeaderToWorksheet(ByVal HeaderArrayPtr As Long, ByRef TopLeftCell As Range)`

Beschreibung: Diese Prozedur überträgt die HeaderMatrix-Struktur als Zeile in ein Excel-Tabellenblatt. Durch den Parameter TopLeftCell (Range) wird dabei die obere, linke Zelle innerhalb des Tabellenblattes zur Abbildung der Überschriftenzeile festgelegt. Der Parameter HeaderArrayPtr (Long) ist dabei ein Zeiger auf die HeaderMatrix-Struktur von PMAktuar (PEMatSpez).

PutMatrixColumnToWorksheet-Prozedur

Syntax: `Public Sub PutMatrixColumnToWorksheet(ByVal ColumnNr As Integer, ByVal TopLeftCell As Range)`

Beschreibung: Diese Prozedur überträgt die durch ColumnNr (Integer) angegebene Spalte der großen Ergebnismatrix in ein Excel-Tabellenblatt. Dem Aufruf dieser Funktion muss der Aufruf einer Funktion vorausgegangen sein, die die große Ergebnismatrix füllt. Durch den Parameter TopLeftCell (Range) wird dabei die obere, linke Zelle innerhalb des Tabellenblattes zur Abbildung der Ergebnismatrix festgelegt.

PutMatrixColumnPartToWorksheetByRow-Prozedur

Syntax: Public Sub PutMatrixColumnPartToWorksheetByRow(ByVal ColumnNr As Integer, ByVal TopLeftCell As Range, ByVal StartFromRow As Integer, Optional ByVal EndAtRow As Integer = 0)

Beschreibung: Diese Prozedur die durch ColumnNr (Integer) angegebene Spalte der großen Ergebnismatrix in ein Excel-Tabellenblatt. Dabei wird nur ein Ausschnitt beginnend von Zeile StartFromRow (Integer) bis zur Zeile EndAtRow (Integer) übertragen. Dem Aufruf dieser Funktion muss der Aufruf einer Funktion vorausgegangen sein, die die große Ergebnismatrix füllt. Durch den Parameter TopLeftCell (Range) wird dabei die obere, linke Zelle innerhalb des Tabellenblattes zur Abbildung der Ergebnismatrix festgelegt. Der optionale Parameter EndAtRow kann ignoriert werden, wenn alle Werte der Spalte bis zum Tafelende zurückgeliefert werden soll, was identisch ist mit der Setzung des Parameters EndAtRow = 0.

PutMatrixColumnPartToWorksheetByAge-Prozedur

Syntax: Public Sub PutMatrixColumnPartToWorksheetByAge(ByVal ColumnNr As Integer, ByVal TopLeftCell As Range, ByVal StartFromAge As Integer, Optional ByVal EndAtAge As Integer = 0)

Beschreibung: Diese Prozedur die durch ColumnNr (Integer) angegebene Spalte der großen Ergebnismatrix in ein Excel-Tabellenblatt. Dabei wird nur ein Ausschnitt beginnend mit den zugehörigen Altern StartFromAge (Integer) bis EndAtAge (Integer) übertragen. Dem Aufruf dieser Funktion muss der Aufruf einer Funktion vorausgegangen sein, die die große Ergebnismatrix füllt. Durch den Parameter TopLeftCell (Range) wird dabei die obere, linke Zelle innerhalb des Tabellenblattes zur Abbildung der Ergebnismatrix festgelegt. Der optionale Parameter EndAtAge kann ignoriert werden, wenn alle Werte der Spalte bis zum Tafelende zurückgeliefert werden sollen, was identisch ist mit der Setzung des Parameters EndAtAge = 0.

Sonstige Berechnungsfunktionen

GetSVRente-Funktion

Syntax: Public Function GetSVRente(ByVal VersJahre As Integer, ByVal JahresGehalt As Double, ByVal NettoAnpassungsfaktor As Double, ByVal BBG As Double) As Double

Beschreibung: (Beschreibung folgt...)

Rückgabewert: String

GetSVRente1998-Funktion

Syntax: Public Function GetSVRente1998(ByVal VersJahre As Integer, ByVal JahresGehalt As Double, ByVal NettoAnpassungsfaktor As Double, ByVal BBG As Double, ByVal RZF As Double) As Double

Beschreibung: (Beschreibung folgt...)

Rückgabewert: String

Grundlegende Parameter

Personenstatus

Beschreibung: Die Berechnungsfunktionen für die Teilwerte (GetTW2004, GetTW2005, GetTWmod) bzw. Anwartschaftsbarwerte (GetPUC, GetDBO) erwarten die Angabe des aktuellen Status der Person für die korrekte Interpretation des entsprechenden Personenbestandes sowie der Leistungsvektoren. Vergleiche dazu den Parameter „Status“ der oben genannten Funktionen.

Mögliche Werte:

- | | |
|---|--|
| 0 | Anwärter, |
| 1 | Alters-Rentner, |
| 2 | Invaliden-Rentner, |
| 3 | Hinterbliebenen-Rentner, |
| 4 | Waisen-Rentner, |
| 5 | Technische Rentner, |
| 6 | gesetzl. unverfallbar Ausgeschiedene, |
| 7 | ohne Berechnung, |
| 8 | vertragl. unverfallbar Ausgeschiedene, |
| 9 | Jubiläumsrückstellung. |

Typ der Invalidenleistung (ILTyp)

Beschreibung: Die Berechnungsfunktionen für die Teilwerte (GetTW2004, GetTW2005, GetTWmod) bzw. Anwartschaftsbarwerte (GetPUC, GetDBO) erwarten die Angabe der Art der Invalidenleistung für die korrekte Interpretation der Werte im Invalidenleistungsvektor (regelmäßig L2). Da der Invalidenleistungsvektor lediglich die Startwerte einer im Invaliditätsfall beginnenden Invalidenleistung angeben können, muss zum einen festgelegt werden, ob die dann sofortbeginnende Invalidenrente lebenslang läuft oder lediglich bis zum Pensionsalter. Werte im Invalidenleistungsvektor ab dem Pensionsalter haben im genauen Verständnis des zu Grunde liegenden Bevölkerungsmodells keinerlei Relevanz (Begründung: der Leistungsfall Invalidität kann letztmalig im letzten Jahr vor der Pensionierung, also in PA-1 eintreten).

Hinweis: In diesem Sinne ist der Invalidenrentenvektor in zwei Sonderfällen auf besondere Art und Weise zu interpretieren: der sog. aufgeschobenen Invalidenrente bzw. der Invalidenrente mit Sprung der Leistungshöhe im Pensionsalter (regelmäßig auf die Höhe der Altersleistung). Bei beiden Invalidenleistungsarten werden Werte im Invalidenleistungsvektor ab dem Pensionsalter erwartet und für die Berechnung zwingend herangezogen. Der Benutzer beachte die besondere Behandlung im Fall der aufgeschobenen Invalidenleistung (Modellierung einer Rente der Höhe 1 ab Pensionsalter).

Mögliche Werte:

- | | |
|---|-------------------|
| 0 | lebenslang, |
| 1 | abgekürzt, |
| 2 | aufgeschoben, |
| 3 | mit Sprung im PA. |

Art der Zusage (ZusageArt):

Beschreibung: Die Berechnungsfunktionen für die Teilwerte (GetTW2004, GetTW2005, GetTWmod) bzw. Anwartschaftsbarwerte (GetPUC, GetDBO) erwarten die Angabe der Art der Zusage für die korrekte Berücksichtigung des Finanzierungsbeginnalters.

Mögliche Werte:

- 1 Altzusage (vor dem 1.1.1987),
- 2 Altzusage (bis zum 31.12.2000),
- 3 Neuzusage (ab 1.1.2001 bis 31.12.2008),
- 4 Neuzusage (ab dem 1.1.2009).

Zahlweise der Leistungen

Beschreibung: Die Berechnungsfunktionen für die Teilwerte (GetTW2004, GetTW2005, GetTWmod) bzw. Anwartschaftsbarwerte (GetPUC, GetDBO) sowie bestimmte Barwertfunktionen erwarten die Angabe der Art der (unterjährigen) Zahlweise der Leistungen.

Mögliche Werte:

- 1 jährlich vorschüssig,
- 2 halbjährlich vorschüssig,
- 3 viermonatlich vorschüssig,
- 4 vierteljährlich vorschüssig,
- 6 zweimonatlich vorschüssig,
- 12 monatlich vorschüssig,
- 1 jährlich nachschüssig,
- 2 halbjährlich nachschüssig,
- 3 viermonatlich nachschüssig,
- 4 vierteljährlich nachschüssig,
- 6 zweimonatlich nachschüssig,
- 12 monatlich nachschüssig.

Typ der Versorgungszusage (ZusageForm)

Beschreibung: Die Berechnungsfunktionen für die Teilwerte (GetTW2004, GetTW2005, GetTWmod) bzw. Anwartschaftsbarwerte (GetPUC, GetDBO) erwarten die Angabe des Typs der Versorgungszusage (klassische Leistungszusage, beitragsorientierte Leistungszusage bzw. Beitragszusage mit Mindestleistung). Die Angabe ist insbesondere für die korrekte Behandlung der Leistungsvektoren (ratielle Veränderung der Leistungsvektoren bei den Anwartschaftsbarwertverfahren) bzw. für nötige Bar- und Teilwertvergleiche (bei echten beitragsorientierten Leistungszusagen im Zusammenhang mit der Entgeltumwandlung) innerhalb der Bewertungsverfahren notwendig.

Mögliche Werte:

- 1 klassische Leistungszusage (LZ),
- 2 beitragsorientierte Leistungszusage (boLZ),
- 3 beitragsorientierte Leistungszusage (echte boLZ),
- 4 Beitragszusage mit Mindestleistung (BZML).

Finanzierungsform der Versorgungszusage (FinanzForm)

Beschreibung: Die Berechnungsfunktionen für die Teilwerte (GetTW2004, GetTW2005, GetTWmod) bzw. Anwartschaftsbarwerte (GetPUC, GetDBO) erwarten die Angabe der Art der Finanzierung (arbeitnehmer- bzw. arbeitgeberfinanziert). Die Angabe ist für die korrekte Interpretation und Berücksichtigung der Unverfallbarkeit notwendig sowie bei den arbeitnehmerfinanzierten Versorgungszusagen aus Entgeltumwandlung für die korrekte Angabe des Teilwerts als größerer Wert im Vergleich zwischen klassischem Teilwert sowie dem Barwert der aus den bisherigen Beiträgen bereits ausfinanzierten Leistungen.

Mögliche Werte:

- 1 Arbeitgeberfinanzierung (AG-finanziert),
- 2 Arbeitnehmerfinanzierung (AN-Finanziert/Entgeltumwandlung),
- 3 Mischfinanzierung (AG-/AN-finanziert).