/\* Begin\_Procs taCreateEmployeeTax2020 \*/

if exists (select \* from dbo.sysobjects where id = Object\_id('dbo.taCreateEmployeeTax') and type = 'P')

begin

drop proc dbo.taCreateEmployeeTax

end

go

create procedure dbo.taCreateEmployeeTax

/\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* (c) 2003 Microsoft Business Solutions, Inc.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*

\* PROCEDURE NAME: taCreateEmployeeTax

\*

\* SANSCRIPT NAME: NA

\*

\* PARAMETERS:

\*

\*

\* DESCRIPTION:

\* TA Create Employee Taxes Stored Procedure

\*

\*

\*

\* TABLES:

\* Table Name Access

\* ========== ======

\* UPR00300 Read/Write

\* UPR00100 Read

\* UPR41100 Read

\* UPR41400 Read

\*

\*

\* PROCEDURES CALLED:

\* taUpdateString

\* tasmGetNextNoteIndex

\* taCreateEmployeeTaxPre

\* taCreateEmployeeTaxPost

\*

\*

\* DATABASE:

\* Company

\*

\*

\* RETURN VALUE:

\*

\* 0 = Successful

\* non-0 = Not successful

\*

\* REVISION HISTORY:

\*

\* Date Who Comments

\* ------------- -------- -------------------------------------------------

\* 04/02/02 ggeiszle Initial Creation

\* 01/28/02 jgould Modified Grants original code to get it to R7.0

\* 02/27/04 Jeff Gould Updated to Release 8.0 - No Changes Required

\* 06/16/05 Jeff Gould Updated to Release 9.0 - No Changes Required

\* 09/11/07 Jeff Gould Fixed Bug ID#41786

\* 04/29/12 Tim Satrom Fixed Bug ID#65159

\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*/

@I\_vEMPLOYID char(15), /\* Employee ID - <Required> \*/

@I\_vFDFLGSTS char(6)='', /\* Federal Filing Status - <Optional> - MAR=Married, SINGLE=Single, BLANK=exempt, NRA=Nonresident Alien \*/

@I\_vFEDEXMPT smallint=0, /\* Number of exemptions claimed - <Optional> \*/

@I\_vADFDWHDG numeric(19,5)=0, /\* Additional Federal Withholding - <Optional> \*/

@I\_vESTFEDWH numeric(19,5)=0, /\* Estimated Federal Withholding - <Optional> \*/

@I\_vSTATECD char(2)='', /\* State Code - <Optional> \*/

@I\_vLOCALTAX char(6)='', /\* Local Tax Code - <Optional> \*/

@I\_vW2BF942E tinyint=0, /\* W2 Box for 942 Employee - <Optional> - 0=False&1=True \*/

@I\_vW2BFDCSD tinyint=0, /\* W2 Box for Deceased - <Optional> - 0=False&1=True \*/

@I\_vW2BFDCMP tinyint=0, /\* W2 Box for Deferred Compensation - <Optional> - 0=False&1=True \*/

@I\_vW2BFLREP tinyint=0, /\* W2 Box for Legal Representative - <Optional> - 0=False&1=True \*/

@I\_vW2BFPPLN tinyint=0, /\* W2 Box for Pension Plan - <Optional> - 0=False&1=True \*/

@I\_vW2BFSTEM tinyint=0, /\* W2 Box for Statutory Employee - <Optional> - 0=False&1=True \*/

@I\_vMCRQGEMP tinyint=0, /\* Medicare Qualified Government Employee - <Optional> \*/

@I\_vEICFLGST char(6)='', /\* EIC Filing Status - <Optional> - BLANK = Not Eligible \*/

/\* MAR=Married Both Spouses file Cert, \*/

/\* MAR2=Married without Spouse filing Cert, & \*/

/\* SINGLE=Single/Head of Household \*/

@I\_vNYTXDiff tinyint=0, /\* Withhold NY Tax Difference - <Optional> \*/

@I\_vUpdateIfExists tinyint=1, /\* Flag to allow Employe Tax data to be updated if it exists. 0=no, 1=yes \*/

@I\_vRequesterTrx smallint=0, /\* Requester Transaction <Optional> - 0=false,1=true - if true than populates Requester shadow table \*/

@I\_vUSRDEFND1 char(50) = '', /\* User Defined field - developer use only \*/

@I\_vUSRDEFND2 char(50) = '', /\* User Defined field - developer use only \*/

@I\_vUSRDEFND3 char(50) = '', /\* User Defined field - developer use only \*/

@I\_vUSRDEFND4 varchar(8000) = '', /\* User Defined field - developer use only \*/

@I\_vUSRDEFND5 varchar(8000) = '', /\* User Defined field - developer use only \*/

@O\_iErrorState int output, /\*Return value: 0=No Errors, 1=Error Occurred \*/

@oErrString varchar(255) output /\*Return Error Code List \*/

with encryption

as

set transaction isolation level read uncommitted

set nocount on

declare

@sCompanyID int,

@iCustomState int,

@iStatus int,

@O\_oErrorState int,

@nNextNoteIndex numeric(19,5), /\* Stores the next note index \*/

@iGetNextNoteIdxErrState int, /\* Return code from call to proc to get next note index \*/

@iError int, /\* Error code from system calls \*/

@DBName char(50), /\* Database name of the company \*/

@iAddCodeErrState int

select

@O\_iErrorState = 0, /\* Set error state to 0 (no errors found)\*/

@O\_oErrorState = 0,

@sCompanyID = CMPANYID from DYNAMICS..SY01500 where INTERID = db\_name()

if (@oErrString is NULL)

begin

select @oErrString = ''

end

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Call Pre Proc \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

exec @iStatus = taCreateEmployeeTaxPre

@I\_vEMPLOYID output,

@I\_vFDFLGSTS output,

@I\_vFEDEXMPT output,

@I\_vADFDWHDG output,

@I\_vESTFEDWH output,

@I\_vSTATECD output,

@I\_vLOCALTAX output,

@I\_vW2BF942E output,

@I\_vW2BFDCSD output,

@I\_vW2BFDCMP output,

@I\_vW2BFLREP output,

@I\_vW2BFPPLN output,

@I\_vW2BFSTEM output,

@I\_vMCRQGEMP output,

@I\_vEICFLGST output,

@I\_vNYTXDiff output,

@I\_vUpdateIfExists output,

@I\_vRequesterTrx output,

@I\_vUSRDEFND1 output,

@I\_vUSRDEFND2 output,

@I\_vUSRDEFND3 output,

@I\_vUSRDEFND4 output,

@I\_vUSRDEFND5 output,

@O\_iErrorState = @iCustomState output,

@oErrString = @oErrString /\*@iCustomErrString\*/ output

select @iError = @@error

if @iStatus = 0 and @iError <> 0

begin

select @iStatus = @iError

end

if (@iStatus <> 0) or (@iCustomState <> 0)

begin

select @O\_iErrorState = 4096 /\* Pre Custom Business Logic for taCreateEmployeeTaxPre returned an error value \*/ /\*EMPLOYID\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@O\_oErrorState output

return (@O\_iErrorState)

end

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Validate input parameters for null \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Verify input parameters are not null \*/

if (

@I\_vEMPLOYID is NULL or

@I\_vFDFLGSTS is NULL or

@I\_vFEDEXMPT is NULL or

@I\_vADFDWHDG is NULL or

@I\_vESTFEDWH is NULL or

@I\_vSTATECD is NULL or

@I\_vLOCALTAX is NULL or

@I\_vW2BF942E is NULL or

@I\_vW2BFDCSD is NULL or

@I\_vW2BFDCMP is NULL or

@I\_vW2BFLREP is NULL or

@I\_vW2BFPPLN is NULL or

@I\_vW2BFSTEM is NULL or

@I\_vMCRQGEMP is NULL or

@I\_vEICFLGST is NULL or

@I\_vNYTXDiff is NULL or

@I\_vUpdateIfExists is NULL or

@I\_vRequesterTrx is NULL or

@I\_vUSRDEFND1 is NULL or

@I\_vUSRDEFND2 is NULL or

@I\_vUSRDEFND3 is NULL or

@I\_vUSRDEFND4 is NULL or

@I\_vUSRDEFND5 is NULL

)

begin

/\* input variable contains null value \*/

select @O\_iErrorState = 4594 /\* A null was found in at least one input parameter for taCreateEmployeeTax \*/ /\*EMPLOYID\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

end

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Master Record Upper Case \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

select @I\_vEMPLOYID =UPPER(@I\_vEMPLOYID),

@I\_vFDFLGSTS = UPPER(@I\_vFDFLGSTS),

@I\_vEICFLGST = UPPER(@I\_vEICFLGST),

@I\_vSTATECD = UPPER(@I\_vSTATECD),

@I\_vLOCALTAX = UPPER(@I\_vLOCALTAX)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Validate Required Fields\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Validate the Employee ID is not blank \*/

if ( @I\_vEMPLOYID = '' )

begin

select @O\_iErrorState = 1569 /\* The Employee ID (EMPLOYID) is blank \*/ /\*EMPLOYID\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

end

/\* Validate the Employee ID actually exists in the UPR00100 table \*/

if ( @I\_vEMPLOYID <> '' )

begin

if (not exists (select 1 from UPR00100 (nolock) where EMPLOYID = @I\_vEMPLOYID ))

begin

select @O\_iErrorState = 1570 /\* The Employee ID (EMPLOYID) passed does not exist in the Employee Master table - UPR00100 \*/ /\*EMPLOYID\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

end

end

/\* Validate the Number of Exemptions <> < 0 \*/

if ( @I\_vFEDEXMPT < 0 )

begin

select @O\_iErrorState = 1571 /\* The Number of Exemptions (FEDEXMPT) can not be < 0 \*/ /\*EMPLOYID\*/ /\*FEDEXMPT\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

end

/\* Validate Estimated Federal Withholding is not < 0 \*/

if ( @I\_vESTFEDWH < 0 )

begin

select @O\_iErrorState = 1572 /\* The Estimated Federal Withholding (ESTFEDWH) can not be < 0 \*/ /\*EMPLOYID\*/ /\*ESTFEDWH\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

end

/\* Validate the Tax Withholding State is a valid value \*/

if ( @I\_vSTATECD <> '' )

begin

if (not exists (select 1 from UPR41100 (nolock) where STATECD = @I\_vSTATECD ))

begin

select @O\_iErrorState = 1573 /\* The State (STATECD) passed does not exist in the Payroll State Code Setup Table - UPR41100 \*/ /\*EMPLOYID\*/ /\*STATECD\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

end

end

/\* Validate the Local Tax passed exists \*/

if ( @I\_vLOCALTAX <> '' )

begin

if (not exists (select 1 from UPR41400 (nolock) where LOCALTAX = @I\_vLOCALTAX ))

begin

select @O\_iErrorState = 1574 /\* The Local Tax (LOCALTAX) passed does not exist in the Payroll Local Tax Setup Table - UPR414100 \*/ /\*EMPLOYID\*/ /\*LOCALTAX\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

end

end

/\* Validate W2 Box for 942 Employee is a valid value \*/

if ( @I\_vW2BF942E < 0 or @I\_vW2BF942E > 1 )

begin

select @O\_iErrorState = 1575 /\* The W2 Box for 942 Employee (W2BF942E) is invalid - 0 and 1 are valid values \*/ /\*EMPLOYID\*/ /\*W2BF942E\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

end

/\* Validate W2 Box for Deceased Employee is a valid value \*/

if ( @I\_vW2BFDCSD < 0 or @I\_vW2BFDCSD > 1 )

begin

select @O\_iErrorState = 1576 /\* The W2 Box for Deceased Employee (W2BFDCSD) is invalid - 0 and 1 are valid values \*/ /\*EMPLOYID\*/ /\*W2BFDCSD\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

end

/\* Validate W2 Box for Deferred Compensation is valid \*/

if( @I\_vW2BFDCMP < 0 or @I\_vW2BFDCMP > 1 )

begin

select @O\_iErrorState = 1577 /\* W2 Box for Deferred Compensation (W2BFDCMP) is invalid - 0 and 1 are valid values \*/ /\*EMPLOYID\*/ /\*W2BFDCMP\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

end

/\* Validate W2 Box for Legal Rep is valid \*/

if ( @I\_vW2BFLREP < 0 or @I\_vW2BFLREP > 1 )

begin

select @O\_iErrorState = 1578 /\* W2 Box for Legal Rep (W2BFLREP) is invalid - 0 and 1 are valid values \*/ /\*EMPLOYID\*/ /\*W2BFLREP\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

end

/\* Validate W2 Box for Pension Plan is valid \*/

if (@I\_vW2BFPPLN < 0 or @I\_vW2BFPPLN > 1 )

begin

select @O\_iErrorState = 1579 /\* W2 Box for Pension Plan (W2BFPPLN) is invalid - 0 and 1 are valid values \*/ /\*EMPLOYID\*/ /\*W2BFPPLN\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

end

/\* Validate the W2 Box for Statuatory Employee is valid \*/

if ( @I\_vW2BFSTEM < 0 or @I\_vW2BFSTEM > 1 )

begin

select @O\_iErrorState = 1580 /\* W2 Box for Statutory Employee (W2BFSTEM) is invalid - 0 and 1 are valid values \*/ /\*W2BFSTEM\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

end

/\* Validate Medicare Qualified Government Employee is valid \*/

if ( @I\_vMCRQGEMP < 0 or @I\_vMCRQGEMP > 1 )

begin

select @O\_iErrorState = 4088 /\* Medicare Qualified Government Employee (MCRQGEMP) is invalid - 0 and 1 are valid values \*/ /\*EMPLOYID\*/ /\*MCRQGEMP\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

end

/\* Validate Withhold NY Tax is valid \*/

if ( @I\_vNYTXDiff < 0 or @I\_vNYTXDiff > 1 )

begin

select @O\_iErrorState = 4089 /\* Withhold NY Tax (NYTXDiff) is invalid - 0 and 1 are valid values \*/ /\*EMPLOYID\*/ /\*NYTXDiff\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

end

/\* Grant \*/

/\*Federal Filing Status\*/

if ( @I\_vFDFLGSTS <> '' and @I\_vFDFLGSTS <> 'SINGLE' and @I\_vFDFLGSTS <> 'MAR' and @I\_vFDFLGSTS <> 'NRA' and @I\_vFDFLGSTS <> 'HOH' and @I\_vFDFLGSTS <> 'HOHHR' and @I\_vFDFLGSTS <> 'MARHR' and @I\_vFDFLGSTS <> 'SGLHR' and @I\_vFDFLGSTS <> 'NRAHA' )

begin

select @O\_iErrorState = 4094 /\* Federal Filing Status (FDFLGSTS) is invalid \*/ /\*EMPLOYID\*/ /\*FDFLGSTS\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

return (@O\_iErrorState)

end

/\*EIC Filing Status\*/

if ( @I\_vEICFLGST <> '' and @I\_vEICFLGST <> 'MAR' and @I\_vEICFLGST <> 'SINGLE' and @I\_vEICFLGST <> 'MAR2' )

begin

select @O\_iErrorState = 4095 /\* EIC Filing Status (EICFLGST) is invalid \*/ /\*EMPLOYID\*/ /\*EICFLGST\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

return (@O\_iErrorState)

end

/\* End Grant \*/

/\* Validate Update if Exists is a valid value \*/

if ( @I\_vUpdateIfExists < 0 or @I\_vUpdateIfExists > 1)

begin

select @O\_iErrorState = 3690 /\* The UpdateIfExists is invalid - 0=False & 1=True \*/ /\*EMPLOYID\*/ /\*UpdateIfExists\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

end

/\* Validate @I\_vRequesterTrx is a valid value \*/

if ( @I\_vRequesterTrx < 0 or @I\_vRequesterTrx > 1)

begin

select @O\_iErrorState = 3691 /\* The RequesterTrx parameter is invalid - 0=False & 1=True \*/ /\*EMPLOYID\*/ /\*RequesterTrx\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

end

if (@O\_iErrorState <> 0)

return (@O\_iErrorState)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Get the next note index\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

exec @iStatus = DYNAMICS..tasmGetNextNoteIndex

@I\_sCompanyID = @sCompanyID,

@I\_iSQLSessionID = 0,

@I\_noteincrement = 1,

@O\_mNoteIndex = @nNextNoteIndex output,

@O\_iErrorState = @iGetNextNoteIdxErrState output

select @iError = @@error

if ((@iStatus <> 0) or (@iGetNextNoteIdxErrState <> 0) or (@iError <> 0))

begin

select @oErrString = rtrim(@oErrString) + ' ' + ltrim(rtrim(@iGetNextNoteIdxErrState))

select @O\_iErrorState = 4090 /\* Unable to obtain the next note index \*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@O\_oErrorState output

end

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* INSERT DATA \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

if not exists(select 1 from UPR00300 where EMPLOYID = @I\_vEMPLOYID)

begin

insert UPR00300(

NOTEINDX,

EMPLOYID,

EXMFRFED,

FDFLGSTS,

FEDEXMPT,

ADFDWHDG,

ESTFEDWH,

STATECD,

LOCALTAX,

W2BF942E,

W2BFDCSD,

W2BFDCMP,

W2BFLREP,

W2BFPPLN,

W2BFSTEM,

MCRQGEMP,

EICFLGST,

NYTXDiff

)

select

@nNextNoteIndex, /\* NOTEINDX \*/

@I\_vEMPLOYID, /\* EMPLOYID \*/

0, /\* EXMFRFED \*/

@I\_vFDFLGSTS, /\* FDFLGSTS \*/

@I\_vFEDEXMPT, /\* FEDEXMPT \*/

@I\_vADFDWHDG, /\* ADFDWHDG \*/

@I\_vESTFEDWH, /\* ESTFEDWH \*/

@I\_vSTATECD, /\* STATECD \*/

@I\_vLOCALTAX, /\* LOCALTAX \*/

@I\_vW2BF942E, /\* W2BF942E \*/

@I\_vW2BFDCSD, /\* W2BFDCSD \*/

@I\_vW2BFDCMP, /\* W2BFDCMP \*/

@I\_vW2BFLREP, /\* W2BFLREP \*/

@I\_vW2BFPPLN, /\* W2BFPPLN \*/

@I\_vW2BFSTEM, /\* W2BFSTEM \*/

@I\_vMCRQGEMP, /\* MCRQGEMP \*/

@I\_vEICFLGST, /\* EICFLGST \*/

@I\_vNYTXDiff /\* NYTXDiff \*/

if (@@error <> 0)

begin

select @O\_iErrorState = 4091 /\* Unable to insert into the Payroll Tax Information Master Table - UPR00300 \*/ /\*EMPLOYID\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

end

end

else

if ( @I\_vUpdateIfExists = 1 )

begin

update UPR00300 set

FDFLGSTS = @I\_vFDFLGSTS,

FEDEXMPT = @I\_vFEDEXMPT,

ADFDWHDG = @I\_vADFDWHDG,

ESTFEDWH = @I\_vESTFEDWH,

STATECD = @I\_vSTATECD,

LOCALTAX = @I\_vLOCALTAX,

W2BF942E = @I\_vW2BF942E,

W2BFDCSD = @I\_vW2BFDCSD,

W2BFDCMP = @I\_vW2BFDCMP,

W2BFLREP = @I\_vW2BFLREP,

W2BFPPLN = @I\_vW2BFPPLN,

W2BFSTEM = @I\_vW2BFSTEM,

MCRQGEMP = @I\_vMCRQGEMP,

EICFLGST = @I\_vEICFLGST,

NYTXDiff = @I\_vNYTXDiff

where EMPLOYID = @I\_vEMPLOYID

if (@@error <> 0)

begin

select @O\_iErrorState = 4092 /\* Unable to update the Payroll Tax Information Master Table - UPR00300 \*/ /\*EMPLOYID\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@iAddCodeErrState output

end

end

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Call Post Proc \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

exec @iStatus = taCreateEmployeeTaxPost

@I\_vEMPLOYID,

@I\_vFDFLGSTS,

@I\_vFEDEXMPT,

@I\_vADFDWHDG,

@I\_vESTFEDWH,

@I\_vSTATECD,

@I\_vLOCALTAX,

@I\_vW2BF942E,

@I\_vW2BFDCSD,

@I\_vW2BFDCMP,

@I\_vW2BFLREP,

@I\_vW2BFPPLN,

@I\_vW2BFSTEM,

@I\_vMCRQGEMP,

@I\_vEICFLGST,

@I\_vNYTXDiff,

@I\_vUpdateIfExists,

@I\_vRequesterTrx,

@I\_vUSRDEFND1,

@I\_vUSRDEFND2,

@I\_vUSRDEFND3,

@I\_vUSRDEFND4,

@I\_vUSRDEFND5,

@O\_iErrorState = @iCustomState output,

@oErrString = @oErrString /\*@iCustomErrString\*/ output

select @iError = @@error

if @iStatus = 0 and @iError <> 0

begin

select @iStatus = @iError

end

if (@iStatus <> 0) or (@iCustomState <> 0)

begin

select @O\_iErrorState = 4093 /\* Post Custom Business Logic for taCreateEmployeeTaxPost returned an error value \*/ /\*EMPLOYID\*/

exec @iStatus = taUpdateString

@O\_iErrorState,

@oErrString,

@oErrString output,

@O\_oErrorState output

return (@O\_iErrorState)

end

return (@O\_iErrorState)

go

grant execute on dbo.taCreateEmployeeTax to DYNGRP

go

/\* End\_Procs taCreateEmployeeTax \*/