SUBROUTINE UDT\_RECORD\_UPDATE(XML.OUT,USER.LANG,XML.IN,USER.ID)

\*

\*

INCLUDE BLABLA.BP RECORD.MAP ;\*MAPPING OF INCOMING XML

\*

XML.EXT.FILE = 'RECORD.ext' ;\*EXTRACTION RULES FOR INCOMING XML

\*

\*

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

\* Parse information from XML Record \*

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

\*

CALL XML.UNI.PARSER(XML.OUT,XML.IN,XML.EXT.FILE,RECORD.OUT,'')

IF XML.OUT THEN RETURN

\*

\* EXAMPLE

\*

RECORD.ID = RECORD.OUT<REC.ID>

DIRECT.SHIP.DATE = RECORD.OUT<REC.DATEDIRECTSHIPMENT>

RECEIPT.DATE = RECORD.OUT<REC.DATERECEPTION>

CREATED.DATE = RECORD.OUT<REC.DATECREATED>

\*

\* DO YOUR PROCESSING...

\*

XML.OUT := '<success><id>':RECORD.ID:'</id></success>'

\*

RETURN

\*

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

\*CONTENT OF XML.UNI.PARSER

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

SUBROUTINE XML.UNI.PARSER(XML.OUT,XML.IN,XML.EXT.FILE,RECORD.OUT,DEBUG.FLAG)

\*

\*

\* XUM UNIDATA Parser

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

\*

INCLUDE INCLUDE.XML XML.H

\*

XML.OUT = ''

\*

OPEN '','\_XML\_' TO XML ELSE

FILE.NAME = '\_XML\_'

ERR.MSG = "Unable to open file \_XML\_!"

GOSUB 9000 ; \* Error

RETURN

END

OPEN '','XML\_DATA' TO XML\_DATA ELSE

FILE.NAME = 'XML\_DATA'

ERR.MSG = "Unable to open file XML\_DATA!"

GOSUB 9000 ; \* Error

RETURN

END

\*

\* Gennerate Random File

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

LOOP

XML.KEY = DATE():TIME():SYSTEM(12):'.xml'

READU R.XML FROM XML\_DATA,XML.KEY THEN RELEASE XML\_DATA,XML.KEY ELSE EXIT

REPEAT

R.XML = XML.IN

WRITE R.XML ON XML\_DATA,XML.KEY

\*

\* Prepare XML

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

IF DEBUG.FLAG THEN PRINT 'PrepareXML started: ':OCONV(TIME(),'MTS')

\*

XML.STATUS = PREPAREXML(FILEINFO(XML\_DATA,2):'/':XML.KEY,XML.HANDLE)

IF XML.STATUS = XML.ERROR THEN

ERR.STATUS = XMLERROR(XML.ERR.MSG)

ERR.MSG = 'Cannot not prepare XML ':XML.ERR.MSG

GOSUB 9000

RETURN

END

\*

\* Get the ORDER information

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

IF DEBUG.FLAG THEN PRINT 'OpenXMLData for order started: ':OCONV(TIME(),'MTS')

\*

XML.STATUS = OPENXMLDATA(XML.HANDLE,FILEINFO(XML,2):'/':XML.EXT.FILE,ORDER.XML.DATA)

IF XML.STATUS = XML.ERROR THEN

ERR.STATUS = XMLERROR(XML.ERR.MSG)

ERR.MSG = 'Cannot not open XML ':XML.ERR.MSG

GOSUB 9000

RETURN

END

\*

IF XML.STATUS = XML.INVALID.HANDLE THEN

ERR.STATUS = XMLERROR(XML.ERR.MSG)

ERR.MSG = 'Invalid XML Handle ':XML.ERR.MSG

GOSUB 9000

RETURN

END

\*

\* Read XML Info

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

IF DEBUG.FLAG THEN PRINT 'ReadXMLData loop for order started: ':OCONV(TIME(),'MTS')

\*

REC.PROC.CNT = 0

LOOP

XML.STATUS = READXMLDATA(ORDER.XML.DATA,RECORD.OUT)

IF XML.STATUS = XML.ERROR THEN

ERR.STATUS = XMLERROR(XML.ERR.MSG)

ERR.MSG = 'Error with ReadXMLData ':XML.ERR.MSG

GOSUB 9000

RETURN

END ELSE

IF XML.STATUS = XML.EOF THEN EXIT

END

REPEAT

\* Close XML

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

XML.STATUS = CLOSEXMLDATA(XML.HANDLE)

IF XML.STATUS = XML.ERROR THEN

ERR.STATUS = XMLERROR(XML.ERR.MSG)

ERR.MSG = 'Error with CloseXMLData ':XML.ERR.MSG

GOSUB 9000

RETURN

END

\*

\* Release XML

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

XML.STATUS = RELEASEXML(XML.HANDLE)

IF XML.STATUS = XML.ERROR THEN

ERR.STATUS = XMLERROR(XML.ERR.MSG)

ERR.MSG = 'Error with ReleaseXML ':XML.ERR.MSG

GOSUB 9000

RETURN

END

\*

\* Delete Temp File

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

DELETE XML\_DATA,XML.KEY

\*

RETURN

\*

\*

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

\* Error \*

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

\*

9000 NULL

\*

CALL XML.FATAL.ERROR(ERR.MSG,'XML.UNI.PARSER',1,XML.OUT)

\*

RETURN

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

\*

\*CONTENT OF .EXT FILE

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

<U2XML\_extraction>

<file\_extraction start = "/Record" />

<!--field extraction rule in element mode-->

<field\_extraction field = "" path ="ID/text()"/>

<field\_extraction field = "" path ="DateDirectShipment/text()"/>

<field\_extraction field = "" path ="DateReception/text()"/>

<field\_extraction field = "" path ="DateCreated/text()"/>

</U2XML\_extraction>

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

\*CONTENT OF .MAP FILE

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

EQU REC.ID TO 1

EQU REC.DATEDIRECTSHIPMENT TO 2

EQU REC.DATERECEPTION TO 3

EQU REC.DATECREATED TO 4