**TICKET NO: Invoice PO Cost Report - D-Ship Order Lines [#040600]**

Requirement:

->develop one XML based report with this query

->we need .xml file and RTF template

->create new concurrent program with parameters

1.Create one Query for this Columns:

  SELECT xss.oe\_line\_source\_type     oe\_line\_source\_type,  
         xss.creation\_date           line\_creation\_date,  
         xss.organization\_code,  
         xss.trx\_number              invoice\_number,  
         xss.ar\_create\_date          invoice\_date,  
         xss.invoice\_currency\_code,  
         xss.salesrep\_name,  
         xss.account\_number,  
         xss.cust\_name,  
         xss.order\_number,  
         xss.line\_number,  
         xss.mfg\_code,  
         xss.mfg\_desc,  
         xss.item\_number,  
         xss.description,  
         xss.quantity\_invoiced,  
         xss.uom\_code,  
         xss.unit\_selling\_price,  
         xss.extended\_amount,  
         xss.std\_unit\_cost           std\_cost,  
         xss.extend\_cost             std\_extended\_cost,  
         xss.mfg\_rebate\_amt,  
         xss.mfg\_rebate\_amt\_extend,  
         xss.special\_rebate\_amt,  
         xss.MFG\_REBATE\_AMT\_EXTEND,  
         --Gross\_Margin  
         xss.unit\_cost               std\_unit\_cost,  
         xss.term\_name,  
         xss.line\_category\_code,  
         xss.mfg\_program\_desc,  
         xss.partner\_authorization,  
         xss.partner\_level,  
         xss.special\_rebate\_descr,  
         xss.mfg\_price\_exception\_num,  
         xss.po\_number,  
         pol.line\_num                po\_line\_number,  
         xss.price\_level             po\_cost,  
         --po\_cost\_ext  
         api.invoice\_num             ap\_invoice,  
         apil.line\_number            ap\_invoice\_line,  
         api.invoice\_amount          ap\_invoice\_cost  
    --Ap\_invoices\_cost\_Ext  
    FROM oe\_drop\_ship\_sources     oed,  
         apps.xbsi\_sales\_summary  xss,  
         po\_headers\_all           poh,  
         po\_lines\_all             pol,  
         apps.ap\_invoice\_lines\_all apil,  
         ap\_invoices\_all          api  
   WHERE     1 = 1  
         AND oed.line\_id = xss.line\_id  
         AND oed.header\_id = xss.header\_id  
         AND poh.po\_header\_id = pol.po\_header\_id  
         AND oed.po\_header\_id = poh.po\_header\_id  
         AND oed.po\_line\_id = pol.po\_line\_id  
         AND pol.po\_header\_id = apil.po\_header\_id  
         AND pol.po\_line\_id = apil.PO\_LINE\_ID  
         AND apil.invoice\_id = api.invoice\_id  
         AND xss.order\_number = 14254087  
ORDER BY xss.creation\_date DESC;

STEP2:

Create one XML file for this Query:

<?xml version="1.0" encoding="UTF-8"?>

<dataTemplate>

<parameters>

<parameter name="p\_mfg\_code" datatype="character" defaultValue=""/>

<parameter name="p\_start\_date" datatype="DATE" defaultValue=""/>

<parameter name="p\_end\_date" datatype="DATE" defaultValue=""/>

<parameter name="p\_bs\_org" datatype="character" defaultValue=""/>

</parameters>

<dataQuery>

<sqlStatement name="G\_PO\_DATA">

<![CDATA[

SELECT xss.OPERATING\_UNIT\_ID,

xss.oe\_line\_source\_type oe\_line\_source\_type,

xss.creation\_date line\_creation\_date,

xss.organization\_code,

xss.trx\_number invoice\_number,

xss.ar\_create\_date invoice\_date,

xss.invoice\_currency\_code,

xss.salesrep\_name,

xss.account\_number,

xss.cust\_name,

xss.order\_number,

xss.line\_number,

xss.mfg\_code,

xss.mfg\_desc,

xss.item\_number,

xss.description,

xss.quantity\_invoiced,

xss.uom\_code,

xss.unit\_selling\_price,

xss.extended\_amount,

xss.std\_unit\_cost std\_cost,

xss.extend\_cost std\_extended\_cost,

xss.mfg\_rebate\_amt,

xss.mfg\_rebate\_amt\_extend,

xss.special\_rebate\_amt,

xss.MFG\_REBATE\_AMT\_EXTEND,

NULL Gross\_Margin,

xss.unit\_cost std\_unit\_cost,

xss.term\_name,

xss.line\_category\_code,

xss.mfg\_program\_desc,

xss.partner\_authorization,

xss.partner\_level,

xss.special\_rebate\_descr,

xss.mfg\_price\_exception\_num,

xss.po\_number,

pol.line\_num po\_line\_number,

xss.price\_level po\_cost,

NULL po\_cost\_ext,

api.invoice\_num ap\_invoice,

apil.line\_number ap\_invoice\_line,

api.invoice\_amount ap\_invoice\_cost,

NULL ap\_invoices\_cost\_Ext

FROM oe\_drop\_ship\_sources oed,

apps.xbsi\_sales\_summary xss,

po\_headers\_all poh,

po\_lines\_all pol,

apps.ap\_invoice\_lines\_all apil,

ap\_invoices\_all api

WHERE 1 = 1

AND oed.line\_id = xss.line\_id

AND oed.header\_id = xss.header\_id

AND poh.po\_header\_id = pol.po\_header\_id

AND oed.po\_header\_id = poh.po\_header\_id

AND oed.po\_line\_id = pol.po\_line\_id

AND pol.po\_header\_id = apil.po\_header\_id

AND pol.po\_line\_id = apil.PO\_LINE\_ID

AND apil.invoice\_id = api.invoice\_id

AND xss.mfg\_code = :p\_mfg\_code

AND TRUNC (xss.ar\_create\_date) >= TRUNC ( :p\_start\_date)

AND TRUNC (xss.ar\_create\_date) <=

(CASE

WHEN :p\_end\_date IS NULL THEN TRUNC ( :p\_start\_date)

ELSE TRUNC ( :p\_end\_date)

END)

AND xss.OPERATING\_UNIT\_ID IN

(CASE

WHEN UPPER ( :p\_bs\_org) = 'BSUS' THEN 121

WHEN UPPER ( :p\_bs\_org) = 'BSCA' THEN 124

WHEN UPPER ( :p\_bs\_org) = 'BSLA' THEN 165

WHEN UPPER ( :p\_bs\_org) = 'BSMA' THEN 185

ELSE NULL

END)

ORDER BY xss.ar\_create\_date;

]]>

</sqlStatement>

</dataQuery>

<dataStructure>

<group name="G\_INVOICE" source="G\_PO\_DATA">

<element name="OE\_LINE\_SOURCE\_TYPE" value="oe\_line\_source\_type"/>

<element name="LINE\_CREATION\_DATE" value="line\_creation\_date"/>

<element name="ORGANIZATION\_CODE" value="organization\_code"/>

<element name="INVOICE\_NUMBER" value="invoice\_number"/>

<element name="INVOICE\_DATE" value="invoice\_date"/>

<element name="INVOICE\_CURRENCY\_CODE" value="invoice\_currency\_code"/>

<element name="SALESREP\_NAME" value="salesrep\_name"/>

<element name="ACCOUNT\_NUMBER" value="account\_number"/>

<element name="CUST\_NAME" value="cust\_name"/>

<element name="ORDER\_NUMBER" value="order\_number"/>

<element name="LINE\_NUMBER" value="line\_number"/>

<element name="MFG\_CODE" value="mfg\_code"/>

<element name="MFG\_DESC" value="mfg\_desc"/>

<element name="ITEM\_NUMBER" value="item\_number"/>

<element name="DESCRIPTION" value="description"/>

<element name="QUANTITY\_INVOICED" value="quantity\_invoiced"/>

<element name="UOM\_CODE" value="uom\_code"/>

<element name="UNIT\_SELLING\_PRICE" value="unit\_selling\_price"/>

<element name="EXTENDED\_AMOUNT" value="extended\_amount"/>

<element name="STD\_COST" value="std\_cost"/>

<element name="STD\_EXTENDED\_COST" value="std\_extended\_cost"/>

<element name="MFG\_REBATE\_AMT" value="mfg\_rebate\_amt"/>

<element name="MFG\_REBATE\_AMT\_EXTEND" value="mfg\_rebate\_amt\_extend"/>

<element name="GROSS\_MARGIN" value="gross\_margin"/>

<element name="SPECIAL\_REBATE\_AMT" value="special\_rebate\_amt"/>

<element name="UNIT\_COST" value="std\_unit\_cost"/>

<element name="TERM\_NAME" value="term\_name"/>

<element name="LINE\_CATEGORY\_CODE" value="line\_category\_code"/>

<element name="MFG\_PROGRAM\_DESC" value="mfg\_program\_desc"/>

<element name="PARTNER\_AUTHORIZATION" value="partner\_authorization"/>

<element name="PARTNER\_LEVEL" value="partner\_level"/>

<element name="SPECIAL\_REBATE\_DESCR" value="special\_rebate\_descr"/>

<element name="MFG\_PRICE\_EXCEPTION\_NUM" value="mfg\_price\_exception\_num"/>

<element name="PO\_NUMBER" value="po\_number"/>

<element name="PO\_LINE\_NUMBER" value="po\_line\_number"/>

<element name="PO\_COST" value="po\_cost"/>

<element name="PO\_COST\_EXT" value="po\_cost\_ext"/>

<element name="AP\_INVOICE" value="ap\_invoice"/>

<element name="AP\_INVOICE\_LINE" value="ap\_invoice\_line"/>

<element name="AP\_INVOICE\_COST" value="ap\_invoice\_cost"/>

<element name="AP\_INVOICES\_COST\_EXT" value="ap\_invoices\_cost\_Ext"/>

</group>

</dataStructure>

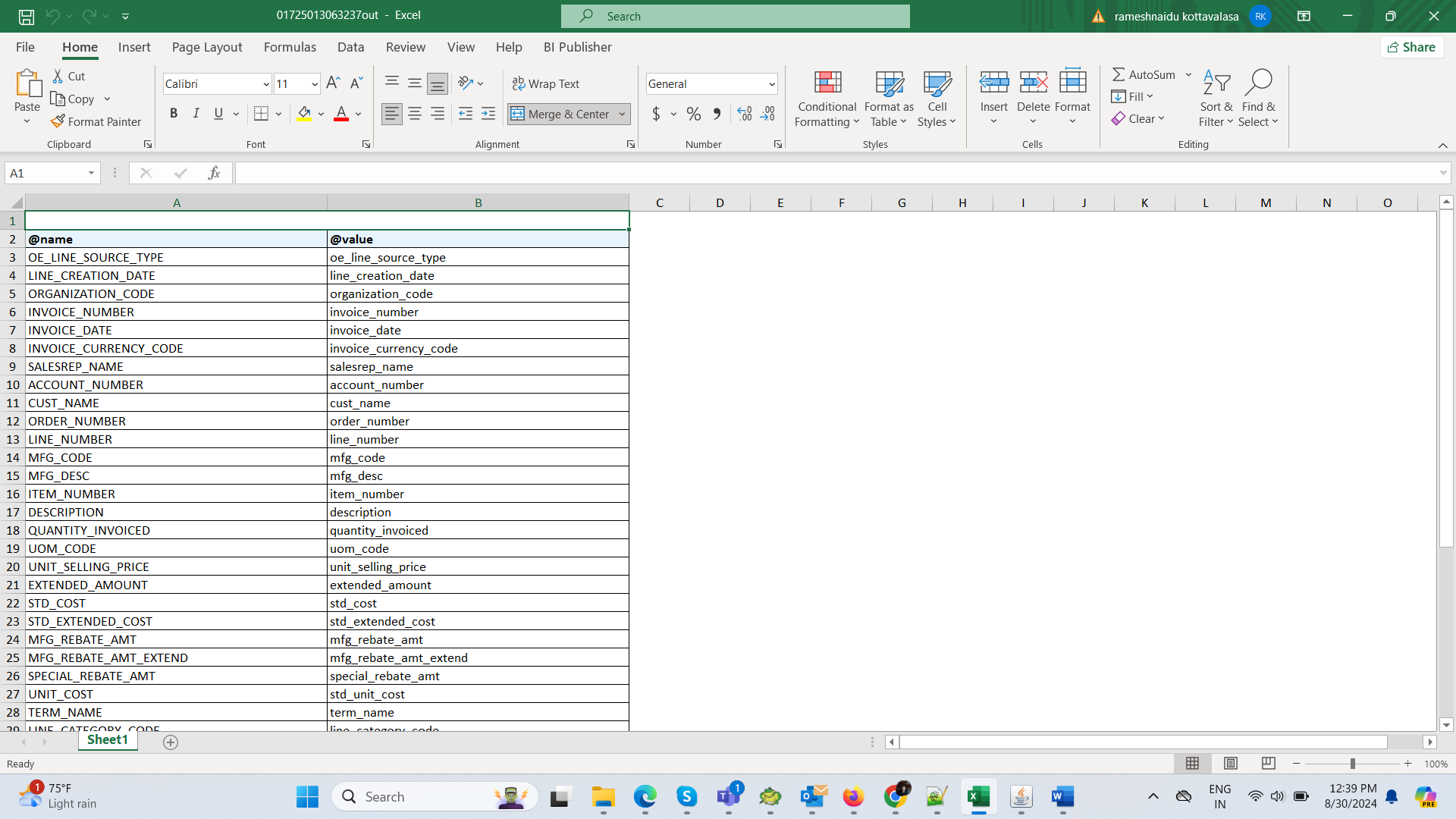
</dataTemplate>

STEP3:

Loading Data Through BI-PUBLISHER:

Open MS Word>Add xml file to sample XML it says Data loaded Successfully

Later Table Wizard open in Excel format you will get data like this.

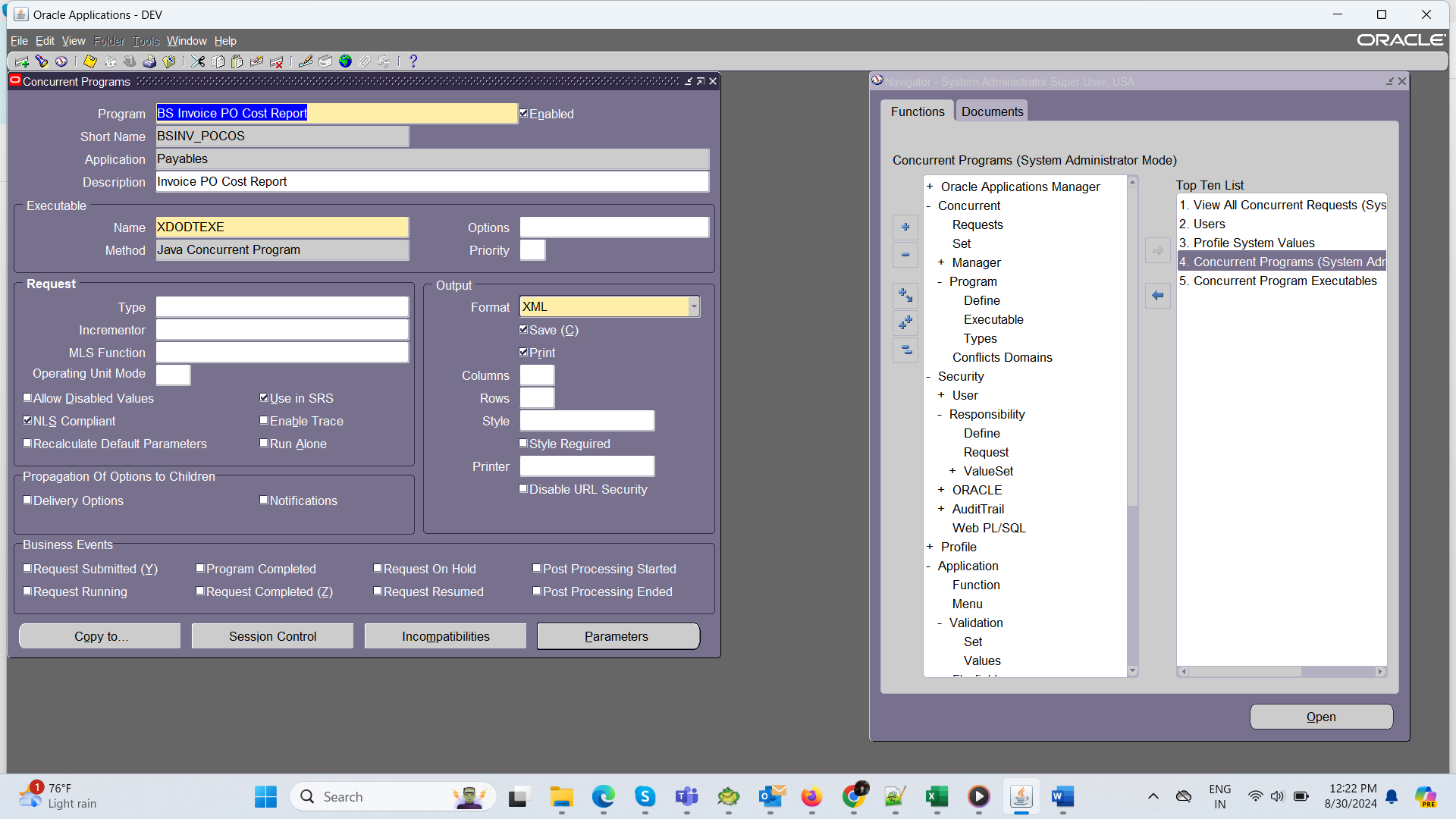


STEP4:

Create one Concurrent Program:

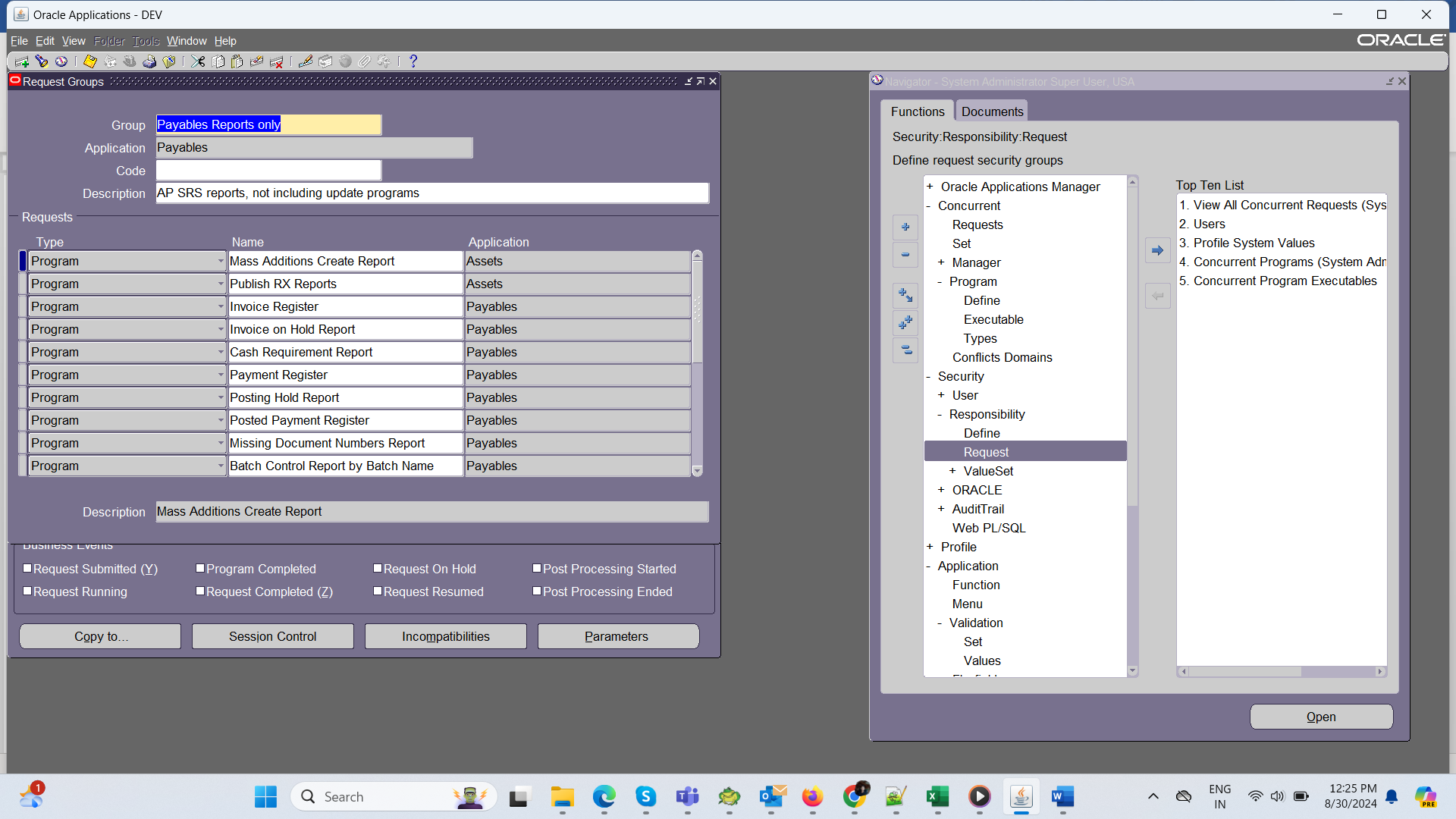
Program\_Name: BS Invoice PO Cost Report

Short\_Name: BSINV\_POCOS

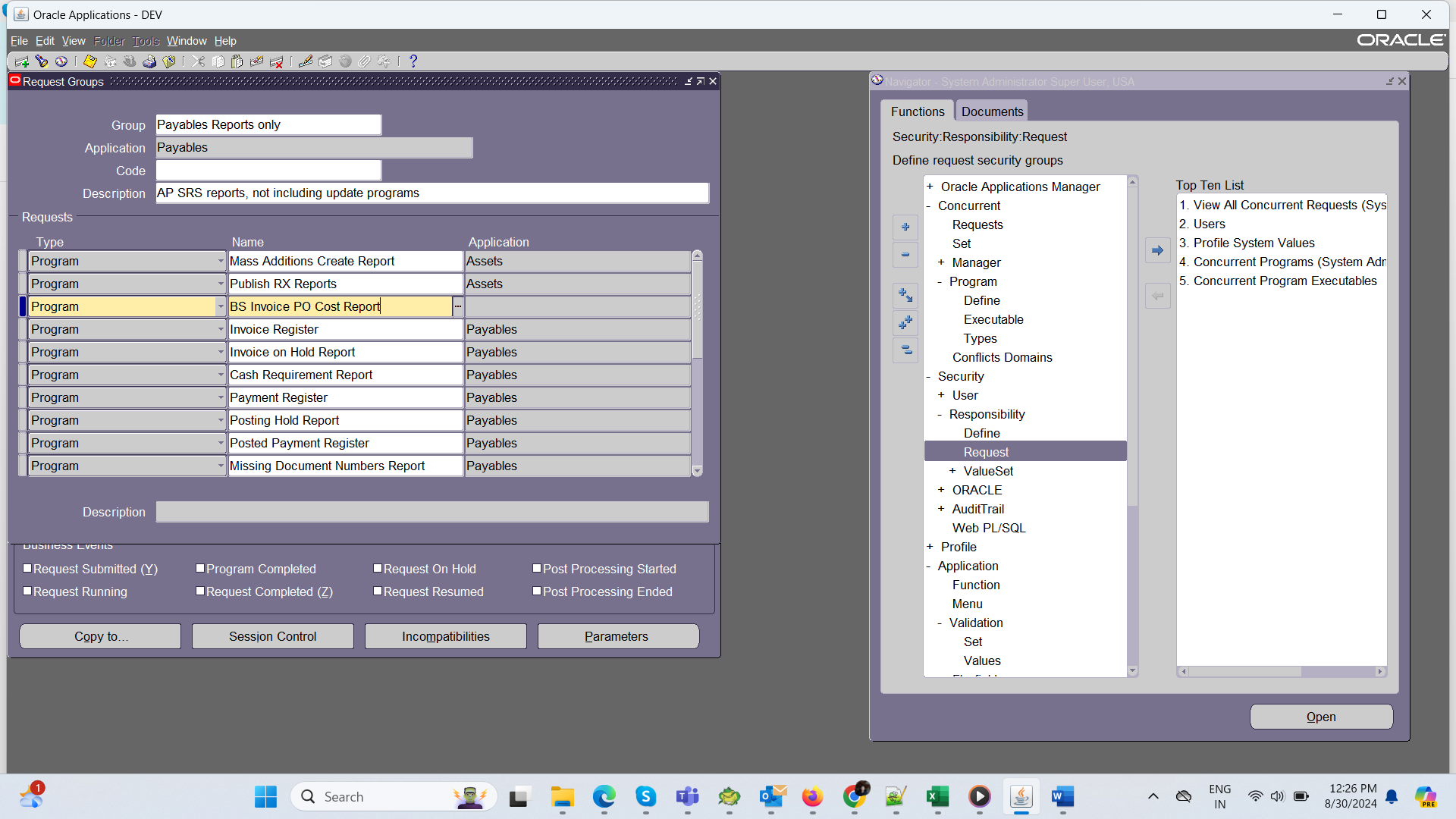


NOTE: Executable name must be ‘XDODTEXE’

Assign this program name to the Responsibility like System administrator>Security>Responsibility>Request>ctrlf11>



Create one new



After creating Concurrent Program Create data definition and data Template.

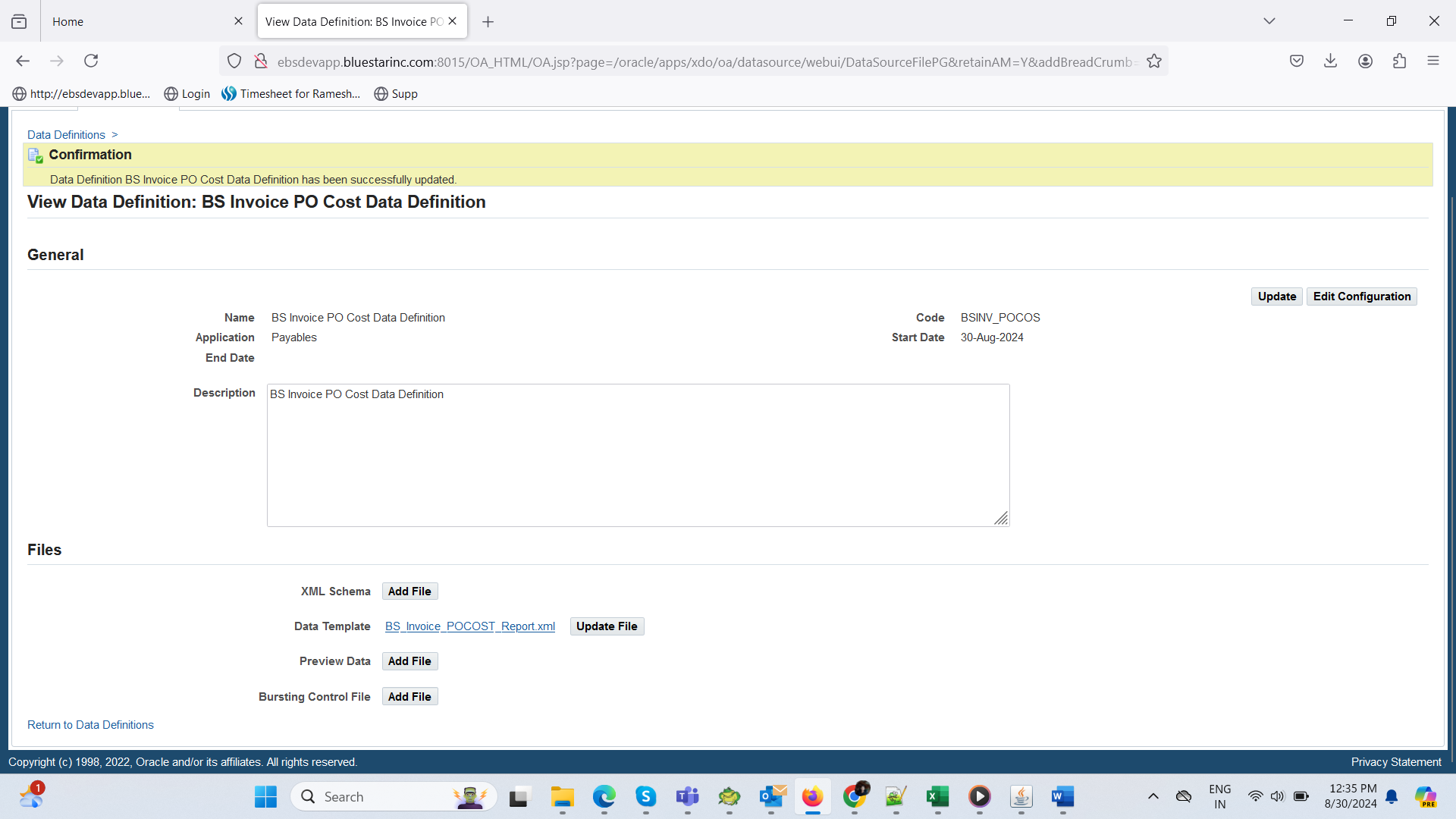
STEP5:

Data Definition:

\*Take Concurrent program short name (copy)

\*Goto XML publisher>Home>Define.

\*paste that short name>go

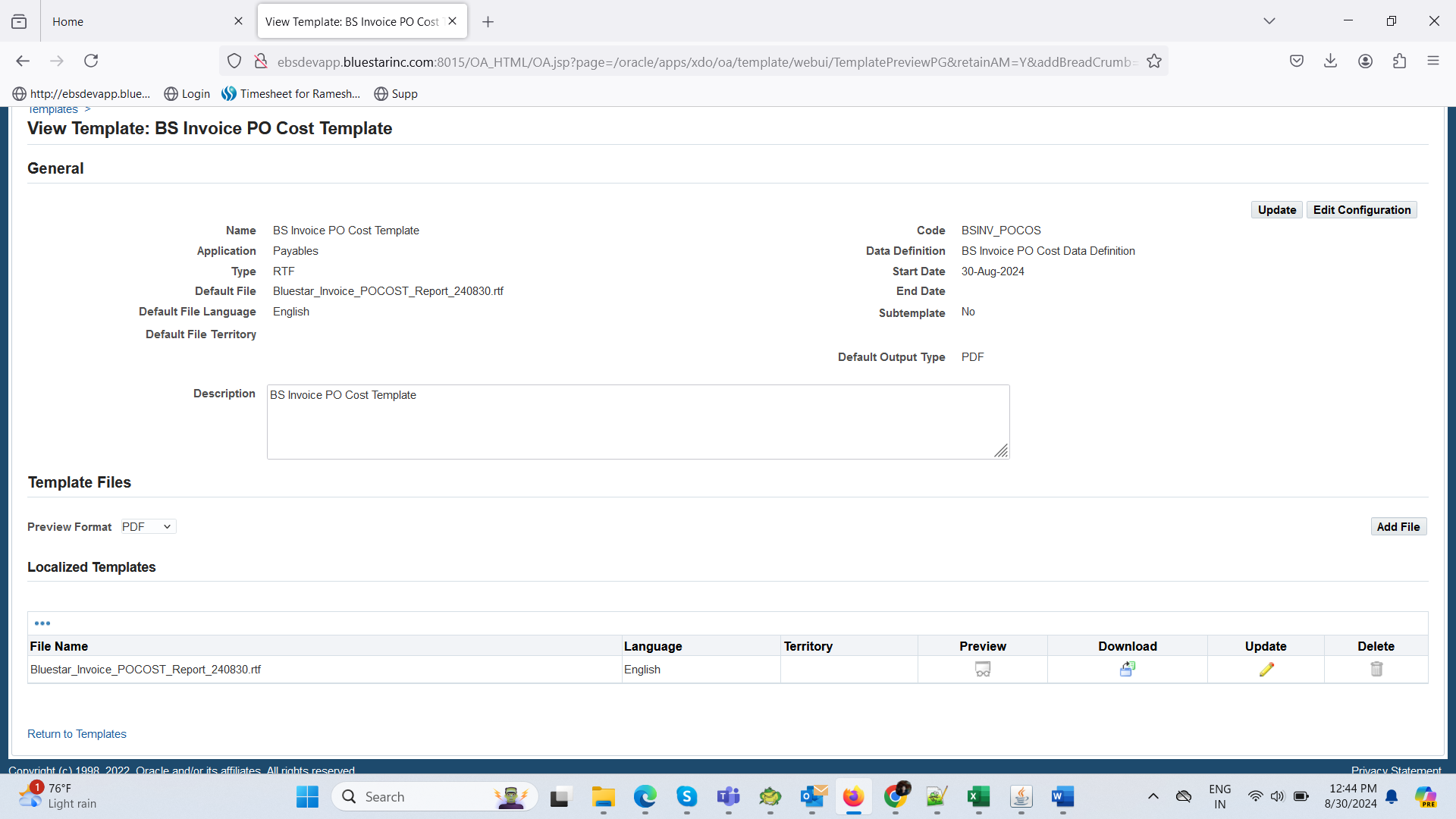


Add XML file in Data Template.

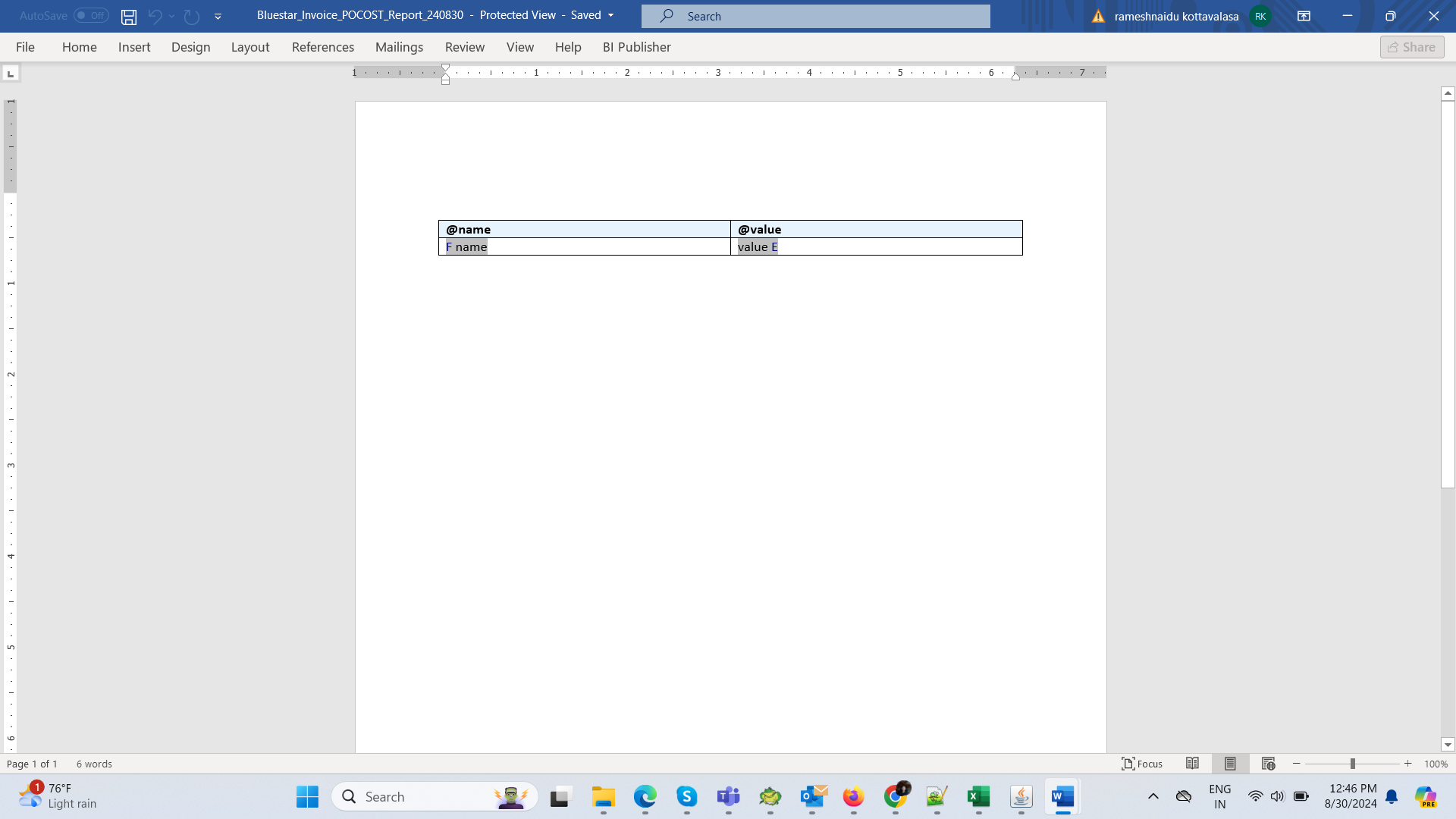
STEP6:

Data Template:

\* Create Data Template like Below

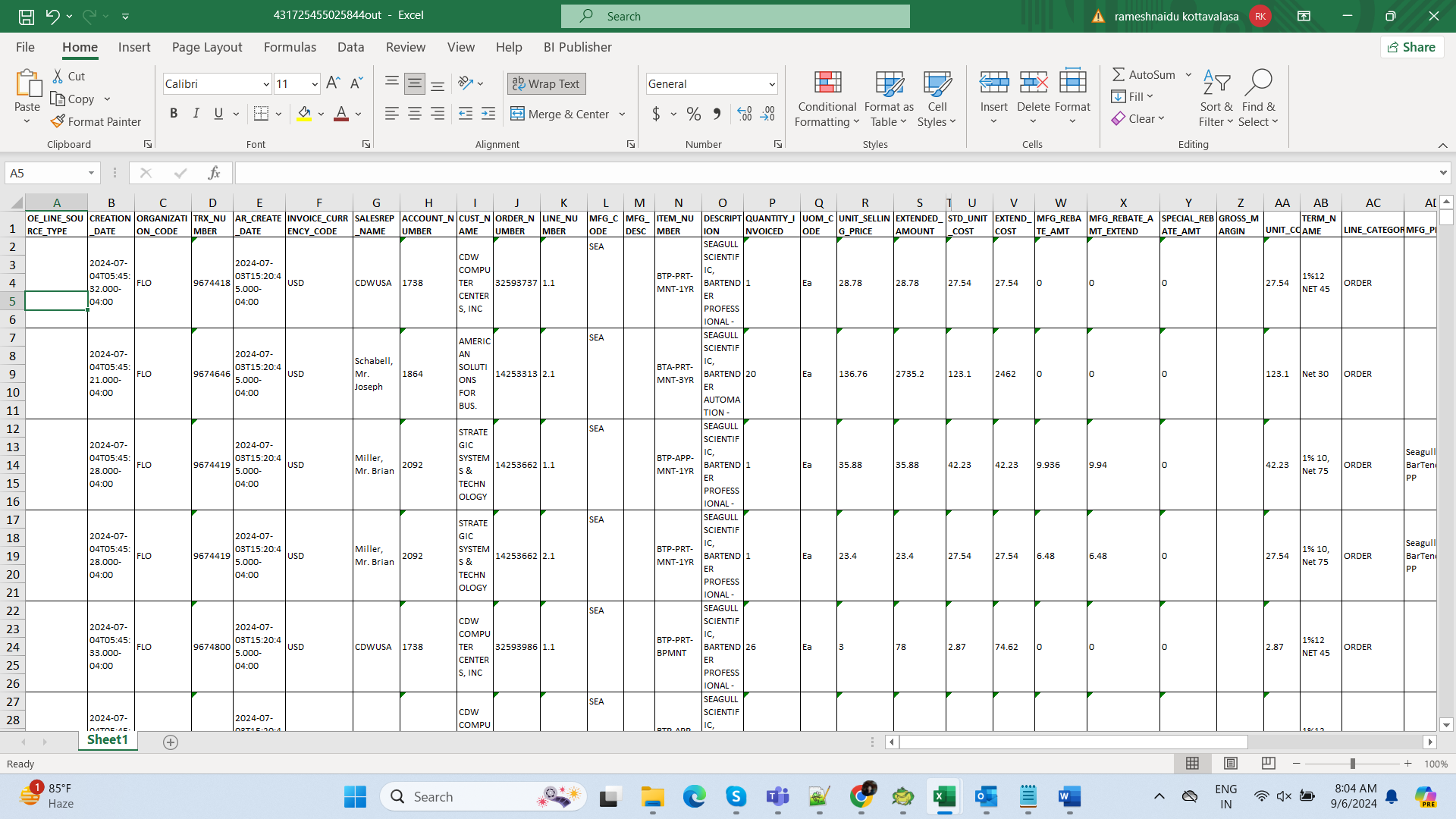


\*Add .RTF file means



\*Add this file in Data Template.

STEP7:

In excel create one excel sheet with columns.  


And bi publisher add xml >open in excel or pdf Formate.