

## AI Agent Studio demo. Using release 25D.

### Agent Team of type “Workflow”. **COMPLEX EXAMPLE.**

#### Scan-read PDF from FTP and generate JSON with LLM to create Sales Order.

##### Introduction/Goal.

We will create this demo from scratch.

We have detailed some steps in previous chapters (OIC integrations and FTP server), but this would be the complete example.

##### **DISCLAIMER.**

**For learning purposes, we use demo Oracle instances for partners, which renew PODS and passwords frequently, as this demo is time consuming we have done by sections, even a part contains screenshots of 26A release which does not affect the explanations.**

**Sorry for that, when 26A is fully available, it will be easier to manage this as export-import will be possible and we will provide the complete code.**

Flow:

1. Create Agent team and trigger parameters: supplier name and file name. (Create Agent team).
2. Read supplier name and get supplier id. (BO).
3. If to check if supplier found with name provided. (IF node).
4. Read PDF file content from FTP server. (REST tool).
5. Create attachment PDF file to supplier. (BO).
6. Get hash code for next step. We will use OIC integration. (REST tool).
- 7. Business Object to read attachment.**
8. Read file from attachment using Document Processor. (Doc. Processor)
9. Use LLM to extract data. (First LLM) and generate purchase requisition payload to use in next creation steps. (outside the scope).
10. Create Sales Order. (BO reused from template provided by Oracle).

**Steps.**

**Steps in Agent Team. AA\_CREATE\_SO\_FROM\_FTP\_PDF (1).**

Agent team Name: AA\_CREATE\_SO\_FROM\_FTP\_PDF.

Description: create sales order from PDF file in FTP server.

AA\_CREATE\_SO\_FROM\_FTP\_PDF

Details

LLM

Security

Triggers

Variables

Error Handling

Agent Team Name

AA\_CREATE\_SO\_FROM\_FTP\_PDF

Agent Team Code

AACREATESOFROMFTPPDF

Family

FIN

Product

Other

Type

Workflow

Maximum Interactions

Description

create sales order from PDF file in FTP server

Include new parameters (Triggers) in Agent Team.

AA\_CREATE\_SO\_FROM\_FTP\_PDF

Details

LLM

Security

Triggers

Variables

Error Handling

Webhook

pSupplier

string

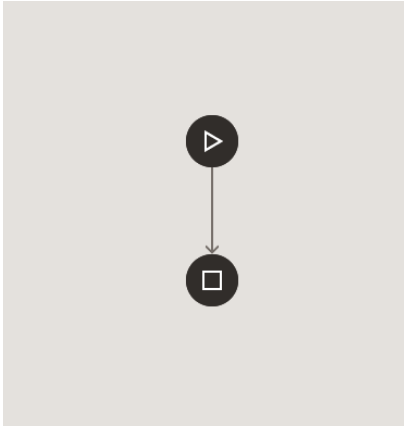
pDirectory

string

pFileName

string

Now we have this:



**Read supplier and get supplier id. Business Object (2).**

Name: AA\_GET\_SUPP\_ID.

Description: get supplier id using supplier name as a parameter.

Note: needed to create and read attachment.

In our instance we have this data to play:

```
{
  "SupplierId": 300000047414503,
  "SupplierPartyId": 300000047414501,
  "Supplier": "Lee Supplies",
```

Edit businessobject - AA\_GET\_SUPP\_ID

Family  
PRC

Name  
AA\_GET\_SUPP\_ID

Resource Type  
Monolith resource

Description  
get supplier id

Product  
Supplier

Code  
ORA\_PRC\_SUPPLIER\_AAGETSUPPID

Resource Path  
/fscmRestApi/resources/11.13.18.05/suppliers

Scenario-function.

Edit scenario - AA\_GET\_SUPP\_ID

Name  
GetSupplierId

Operation Type  
HTTP GET

Description  
Description: get supplier id using supplier name as a parameter

Resource Path  
/fscmRestApi/resources/11.13.18.05/suppliers?q=Supplier={pSupplier}&onlyData=true&fields=SupplierId,SupplierPartyId,Supplier,SupplierNumber

Resource PATH:

```
/fscmRestApi/resources/11.13.18.05/suppliers?q=Supplier='{pSupplier}'&onlyData=true&fields=SupplierId,SupplierPartyId,Supplier,SupplierNumber
```

Name of function: GetSupplierId.

Description: get supplier id using supplier name as a parameter.

Parameter: name of the supplier.

Parameters and Business Object Fields

Add Parameter

Add Field

Add Field from Specification

Name	Data Type	Description	Is Parameter	Actions
pSupplier	string	Supplier	true	<div><div></div><div></div></div>

**Edit parameter/field**

Name

pSupplier

Data Type

String

Description

name of the supplier.

Example.

**Add a new example payload**

Description

Supplier

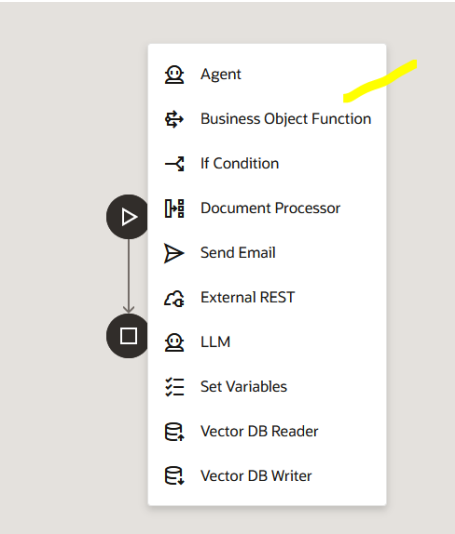
Sample Payload

```
{
  "pSupplier": "Lee Supplies"
}
```

**Steps in Agent Team. AA CREATE SO FROM FTP PDF.**

Added previous Business Object to Agent Team as Business Object Function.

Business Object function name: GET\_SUPP\_ID.



**GET\_SUPP\_ID**  
Business Object Function

Name GET_SUPP_ID	Code GETSUPPID	Error Handler ▼
Family FIN ▼	Product Other ▼	Business object AA_GET_SUPP_ID ▼
Function GetSupplierid ▼		

**Parameters**

pSupplier	string	✎ ▼
-----------	--------	-----

Output.

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "Internal_List1",
  "type": "object",
  "properties": {
    "items": {
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "SupplierId": {
            "type": "integer"
          },
          "SupplierPartyId": {
            "type": "number"
          },
          "SupplierNumber": {
            "type": "string"
          },
          "Supplier": {
            "type": "string"
          }
        }
      }
    }
  }
}
```

## GET\_SUPP\_ID

Business Object Function

### Parameters

### Output

Output Builder Mode

☒ Specification Mode ☐ Simple Mode

Properties

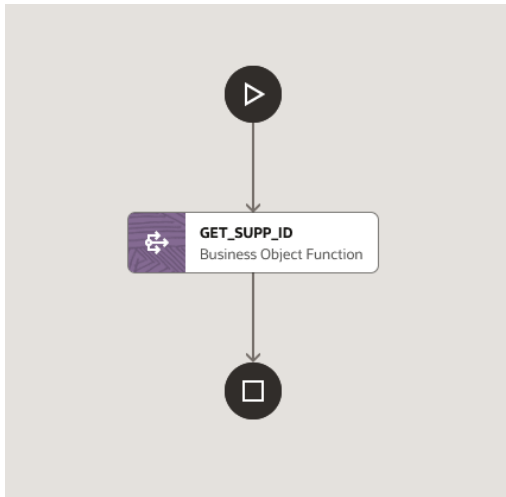
```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "Internal_List1",
  "type": "object",
  "properties": {
    "items": {
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "SupplierId": {
            "type": "integer"
          }
        }
      }
    }
  }
}
```

Parameter and value assigned.

### Parameters

Name  
pSupplier

Value  
{{\$context.\$triggers.REST.\$input.pSupplier}}



Initial testing.

Payload for initial testing:

```
{
  "pDirectory": "/files/Nordea/AP_Invoices/Outbound/LOG",
  "pFileName": "PO_US165029_5.pdf",
  "pSupplier": "Lee Supplies"
}
```

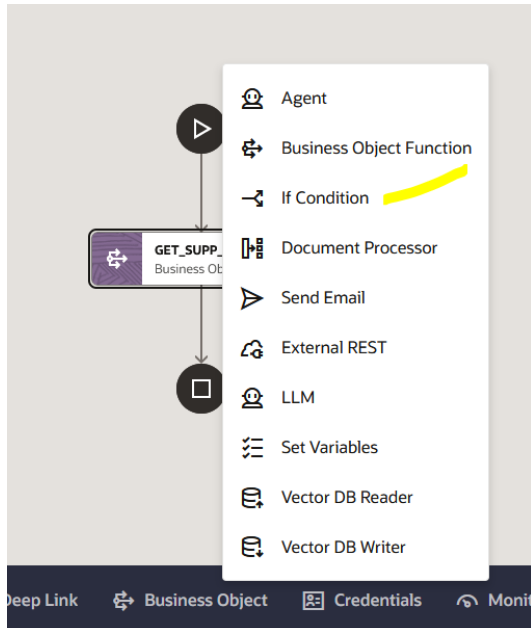
**GET\_SUPP\_ID**

```
{
  "items": [
    {
      "SupplierId": "300000047414503",
      "SupplierPartyId": "300000047414501",
      "Supplier": "Lee Supplies",
      "SupplierNumber": "1252",
      "@context": {
        "key": "300000047414503",
        "headers": {
          "ETag": "ACED0005737200136A6176612E7574696C2E41727261794C6973747881D21D99C7619D03000149000473697A65787000000007770400000007737200116A6176612E6C616E672E496E746567657212E2A0A4F781873802000149000576616C7565787200106A6176612E6C616E672E4E756D62657286AC951D0B94E08B0200007870000001937371007E0002000000457372001B6F7261636C652E6A626F2E646F6D61696E2E4E756C6C56616C75655899C1C58DAABEEB02000149000A6D53514C547970654964787000000000C7371007E0002000000A071007E00077371007E0002000000087371007E00020000000678}}",
          "count": 1,
          "hasMore": false,
          "limit": 25,
          "offset": 0
        }
      }
    }
  ],
  "count": 1,
  "hasMore": false,
  "limit": 25,
  "offset": 0
}
```



### Now let's check if supplier is found with IF node (3).

Name: IF\_SUPP\_FOUND.



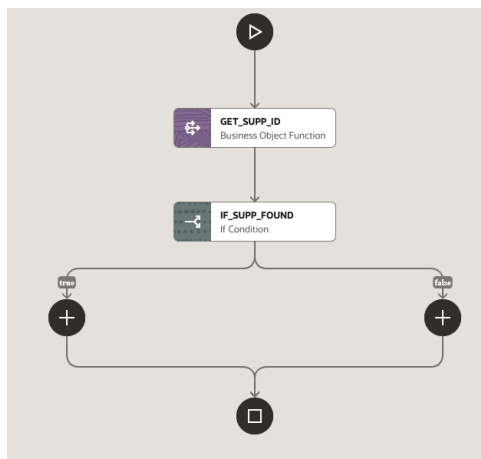
In "If Condition" use "" or null to compare to nulls.

```
{{ $context.$nodes.GETSUPPID.$output.items[0].SupplierId != null }}
```

Other examples:

```
{{ $context.$variables.businessUnitName != '' }}  
{{ $context.$nodes.CREATEORDERSFORPOTOSO.$output.HeaderId != '' }}  
{{ $context.$system.$inputMessage == 'CONVERT_PO_TO_SO' }}
```

Now we have this:





Testing.

Payload for initial testing:

```
{  
  "pDirectory": "/files/Nordea/AP_Invoices/Outbound/LOG",  
  "pFileName": "PO_US165029_0.pdf",  
  "pSupplier": "Lee Supplies"
```

}






**GET\_SUPP\_ID**



Latency  
0.64 seconds


Input:  
{"pSupplier":"Lee Supplies"}

Output:  
{ "items": [{"SupplierId": "300000047414503", "SupplierPartyId": "300000047414501", "Supplier": "Lee Supplies", "SupplierNumber": "1252", "@context": {"key": "300000047414503", "headers": {"Etag": "ACED0005737200136A6176612E7574696C2E41727261794C6973747881D21D99C7619D03000149000473697A65787000000007770400000007737200116A6176612E6C616E672E496E746567657212E2A0A4F781873802000149000576616C7565787200106A6176612E6C616E672E4E756D62657286AC951D0B94E08B0200007870000001937371007E000200000045737200186F7261636C652E6A626F2E646F6D61696E2E4E756C6C56616C75655899C1C58DAABEEB02000149000A6D53514C54797065496478700000000C7371007E0002000000A071007E00077371007E000200000087371007E0002000000678"} }], "count": 1, "hasMore": false, "limit": 25, "offset": 0}

Context:  
 Context

Tools:  
 GET\_SUPP\_ID 

**IF\_SUPP\_FOUND**

 true

**Tool of type External REST (4).**

Name: AA\_READ\_FILE\_FROM\_FTP.

Description: Read file from FTP server.

## AA\_READ\_FILE\_FROM\_FTP

Tool Type  
External REST

Tool Name  
AA\_READ\_FILE\_FROM\_FTP

Tool Code  
AAREADFILEFROMFTP

Family  
FIN

Product  
Other

Description  
Read file from FTP server

OIC integration (provided .iar file): XX\_AI\_FF\_GET\_FILE\_CONTENT (1.0.2)

Purpose: retrieve content of PDF file in base64.

[illegible]

## Authorization.

### Instance URL.

`https://oic3-partners-bots4saas-yu.integration.ca-montreal-1.ocp.oraclecloud.com`

☐ Enable if you want human approval before running the tool.

**Authorization**   Functions

### Authorization

Instance URL  
isaas-yu.integration.ca-montreal-1.ocp.oraclecloud.com

Authentication  
Basic

Description  
get file content from ftp server

Authentication details

User Name  
john.dunbar

Password  
•

Function Endpoint.

Name: get\_content.

Description: get file content in base64.

Resource path URL:

/ic/api/integration/v1/flows/rest/XX\_AI\_FF\_GET\_FILE\_CONTE/1.0/

Operation type: POST.

Body template:

```
{
  "pFileName": "{pFileName}",
  "pDirectory": "{pDirectory}"
}
```

Add Endpoint

Name

get\_content

Operation Type

HTTP POST

Resource Path

/ic/api/integration/v1/flows/rest/XX\_AI\_FF\_GET\_FILE\_CONTE/1.0/

Description

get file content

Body Template

```
{
  "pFileName": "{pFileName}",
  "pDirectory": "{pDirectory}"
}
```

Parameters.

Body Template

```
{
  "pFileName": "{pFileName}",
  "pDirectory": "{pDirectory}"
}
```

Parameters

Sample Queries

Headers

+ Add

Name

pDirectory

Description

pDirectory

Data Type

String

Name

pFileName

Description

Data Type

String

Sample Query.

Parameters Sample Queries Headers

Query

```
{
  "pFileName": "PO_US165029_1.pdf",
  "pDirectory": "/files/XXX/AP_Invoices/Outbound/LOG"
}
```

Description

Payload

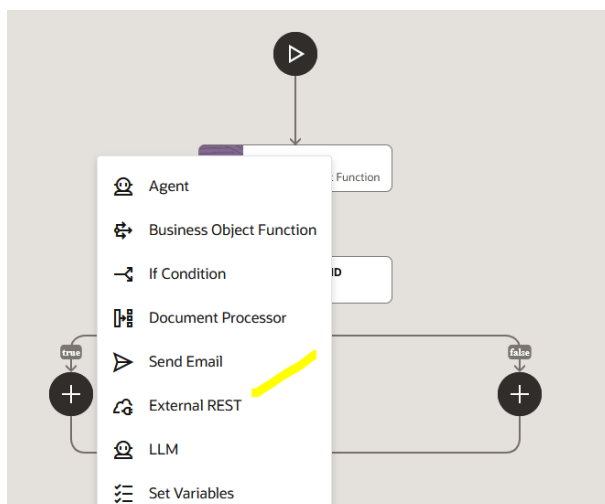
Payload for test:

```
{
  "pDirectory": "/files/Nordea/AP_Invoices/Outbound/LOG",
  "pFileName": "PO_US165029_0.pdf",
  "pSupplier": "Lee Supplies"
}
```

Tool is now completed.

Tools Library			
<div> <div>aa_read</div> <div>×</div> <div>Search by tool name, code, or de...</div> </div> <div>Family Product Tool Type Created By Created On</div>			
1 item <div>+ Add</div>			
Tool Name	Description	Tool Code	Type
AA_READ_FILE_FROM_FTP	get file content	AAREADFILEFROMFTP	External REST

Now in Agent Team add new node of type External REST to include previously created tool.



Name in Agent Team: GET\_FILE\_CONTENT.

Tool name to use: AA\_READ\_FILE\_FROM\_FTP.

GET\_FILE\_CONTENT

External REST

Name

GET\_FILE\_CONTENT

Code

GETFILECONTENT

Error Handler

Family

FIN

Product

Other

Tool Name

AA\_READ\_FILE\_FROM\_FTP

Function

get\_content

Parameters

pDirectory

string

pFileName

string

Output

Output Builder Mode

Specification Mode

Simple Mode

Properties

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "data_base64": {
      "type": "string"
    }
  }
}
```

Output.

Output

Output Builder Mode

Specification Mode

Simple Mode

Properties

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "data_base64": {
      "type": "string"
    }
  }
}
```

Output payload.

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "data_base64": {
      "type": "string"
    }
  }
}
```

Parameters in REST tool and values assigned from trigger variables.

Parameters

Name

pDirectory

Value

{{context.\$triggers.REST.\$input.pDirectory}}

Type {{ to start seeing context-sensitive expressions. }}

Description

Parameters

Name

pFileName

Value

{{context.\$triggers.REST.\$input.pFileName}}

Type {{ to start seeing context-sensitive expressions. }}

Description

Now we have.





## Create attachment PDF file to supplier. Business Object (5).

Goal: creating attachments in suppliers.

Name: AA\_ATTACH\_SUPP.

Description: attach PDF file to supplier.

Resource path:

**Create a new business object**

Cancel Add New Data Source Application

Family: FIN Product: Other

Name: AA\_ATTACH\_SUPP Code: ORA\_FIN\_OTHER\_AAATTACHSUPP

Resource Type: Monolith resource Resource Path: /fscmRestApi/resources/11.13.18.05/suppliers

Oracle Data Source Application Description: attach PDF file to supplier

**Business Object Functions**

+ Add from Specification Import

Name	Description	Resource Path	Actions
attach_pdf_to_supplier	Attach PDF to supplier.	/fscmRestApi/resources/11.13.18.05/suppliers/{id}/child/attachments	

## Scenario-function.

Name: attach\_pdf\_to\_supplier

Description: Attach PDF to supplier.

Operation: POST.

Resource path.

/fscmRestApi/resources/11.13.18.05/suppliers/{id}/child/attachments

## REST API To investigate:

<https://docs.oracle.com/en/cloud/saas/procurement/25d/fapra/op-suppliers-supplierid-child-attachments-post.html>

**Edit scenario - AA\_ATTACH\_SUPP**

Name: attach\_pdf\_to\_supplier Description: Attach PDF to supplier.

Operation Type: HTTP POST Resource Path: /fscmRestApi/resources/11.13.18.05/suppliers/{id}/child/attachments

Use native authentication Header ☒

Operation Payload

```
{
  "DatatypeCode": "FILE",
  "FileName": "{file_name}",
  "CategoryName": "FROM_SUPPLIER",
  "UploadedFileContentType": "application/pdf",
  "ContentRepositoryFileShared": "false",
  "Title": "{title}",
  "Description": "demo",
  "FileContents": "{content}"
}
```

Payload.

```
{
  "DatatypeCode": "FILE",
  "FileName": "{file_name}",
  "CategoryName": "FROM_SUPPLIER",
  "UploadedFileContentType": "application/pdf",
  "ContentRepositoryFileShared": "false",
  "Title": "{title}",
  "Description": "demo",
  "FileContents": "{content}"
}
```

Parameters and Business Object Fields

Add ParameterAdd FieldAdd Field from Specification

Name	Data Type	Description	Is Parameter
id	string	supplier id	true
file_name	string	file name	true
title	string	file title	true
content	string	file content	true

Parameter values:

- Id: supplier id.
- The rest self-explained.

Sample payload.

Example Usages

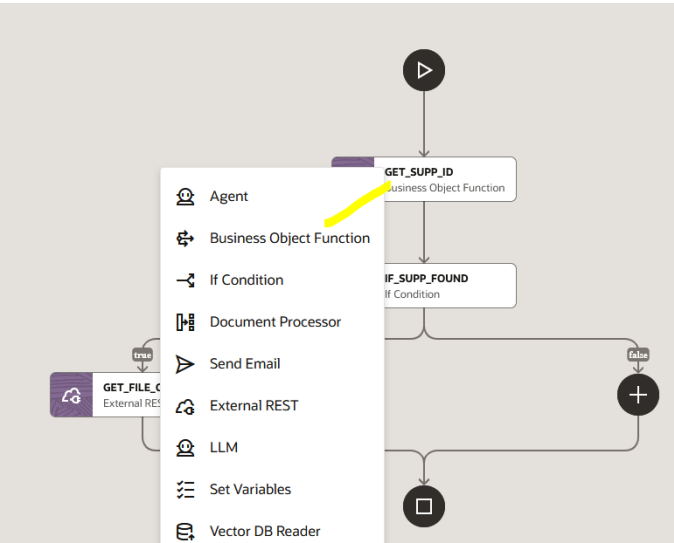
+ Add

Description	Sample Payload	Actions
payload	{ "DatatypeCode": "FILE", "FileName": "file_name1", "CategoryName": "FROM_SUPPLIER", "UploadedFileContentType": "application/pdf", "ContentRepositoryFileShared": "false", "Title": "title1", "Description": "demo", "FileContents": "content1" }	

```
{
  "DatatypeCode": "FILE",
  "FileName": "file_name1",
  "CategoryName": "FROM_SUPPLIER",
  "UploadedFileContentType": "application/pdf",
  "ContentRepositoryFileShared": "false",
  "Title": "title1",
  "Description": "demo",
  "FileContents": "content1"
}
```

**Steps in Agent Team. AA CREATE SO FROM FTP PDF.**

Include Business Object Function in Agent Team.



Name: ATTACH\_SUPP\_FILE.

Using Business Object: AA\_ATTACH\_SUPP.

**ATTACH\_SUPP\_FILE**  
Business Object Function

Name

ATTACH\_SUPP\_FILE

Code

ATTACHSUPPFILE

Error Handler

Family

FIN

Product

Other

Business object

AA\_ATTACH\_SUPP

Function

attach\_pdf\_to\_supplier

Parameters

id	string
file_name	string
title	string
content	string

Value of parameters.

Parameter 1.	Parameter 2.
<div><div>Parameters</div><div><div>Name</div><div>id</div><div>Type</div><div>String</div></div><div><div>Value</div><div>{{context.\$nodes.GETSUPPID.\$output.items[0].SupplierId}}</div><div>Type {} to start seeing context-sensitive expressions.</div></div></div> <div>{{context.\$nodes.GETSUPPID.\$output.items[0].SupplierId}}</div>	<div><div>Name</div><div>title</div><div>Type</div><div>String</div></div> <div><div>Value</div><div>{{context.\$triggers.REST.\$input.pFileName}}</div><div>Type {} to start seeing context-sensitive expressions.</div></div> <div>{{context.\$triggers.REST.\$input.pFileName}}</div>
Parameter 3.	Parameter 4.
<div><div>Name</div><div>title</div><div>Type</div><div>String</div></div> <div><div>Value</div><div>{{context.\$triggers.REST.\$input.pFileName}}</div><div>Type {} to start seeing context-sensitive expressions.</div></div> <div>{{context.\$triggers.REST.\$input.pFileName}}</div>	<div><div>Name</div><div>content</div><div>Type</div><div>String</div></div> <div><div>Value</div><div>{{context.\$nodes.GETFILECONTENT.\$output.data_base64}}</div><div>Type {} to start seeing context-sensitive expressions.</div></div> <div>{{context.\$nodes.GETFILECONTENT.\$output.data_base64}}</div>

Values assigned:

```
id           : {{context.$nodes.GETSUPPID.$output.items[0].SupplierId}}
file_name    : {{context.$triggers.REST.$input.pFileName}}
title        : {{context.$triggers.REST.$input.pFileName}}
content      : {{context.$nodes.GETFILECONTENT.$output.data_base64}}
```

Output.

ATTACH\_SUPP

Business Object Function

Parameters

me\_name

string

title

string

content

string

Output

Output Builder Mode

☒ Specification Mode

☐ Simple Mode

Properties

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "AttachedDocumentId": {
      "type": "integer"
    }
  }
}
```

Output payload.

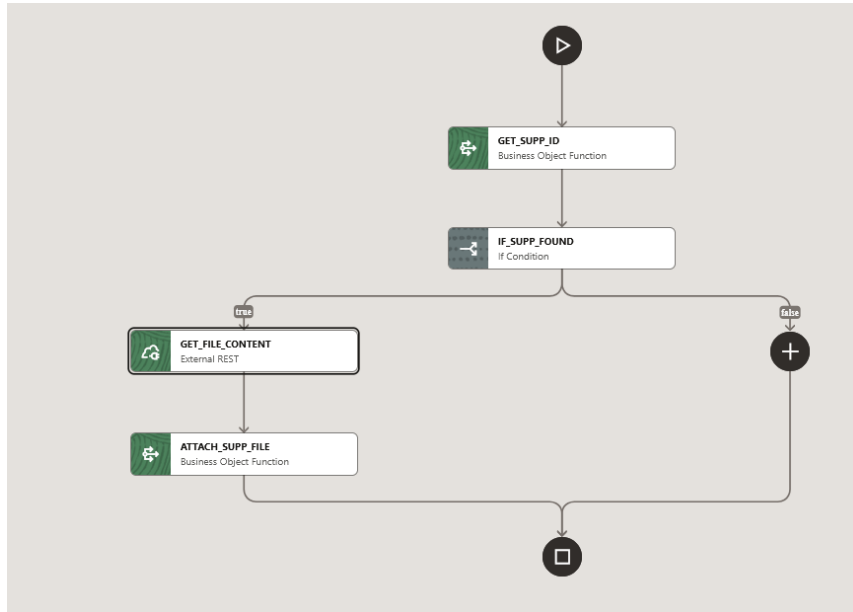
```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "AttachedDocumentId": {
```

```

    "type": "integer"
  }
}

```

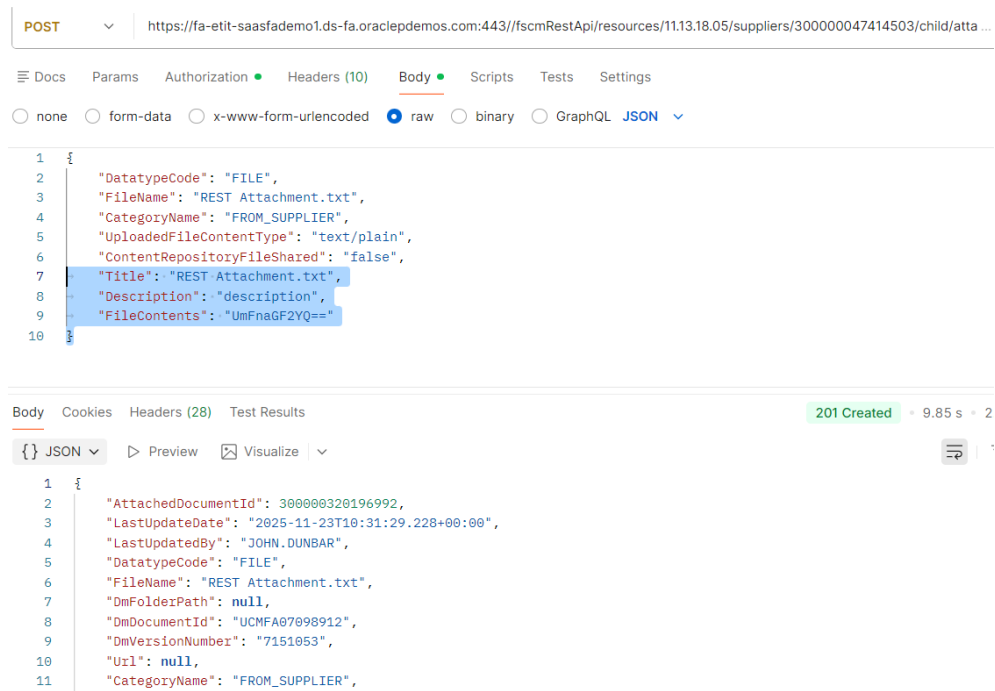
We now have this:



## Note.

Example of payload used in investigation with Postman to attach file to supplier:

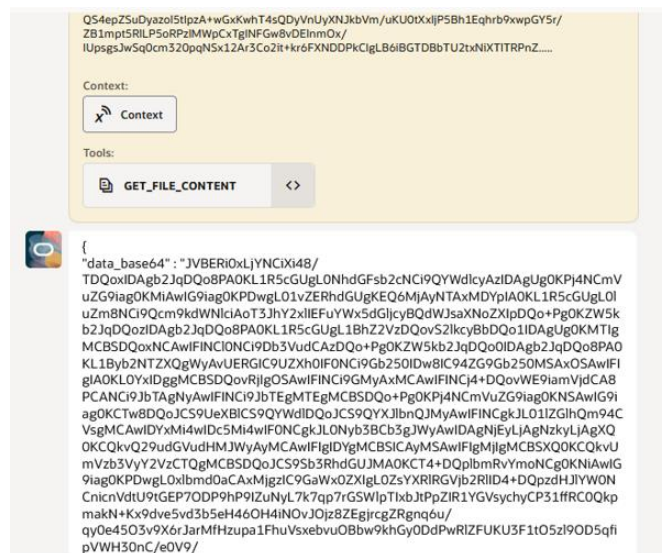
```
{
  "DatatypeCode": "FILE",
  "FileName": "REST Attachment.txt",
  "CategoryName": "FROM_SUPPLIER",
  "UploadedFileContentType": "text/plain", >>>>> "application/pdf",
  "ContentRepositoryFileShared": "false",
  "Title": "REST Attachment.txt",
  "Description": "description",
  "FileContents": "UmFnaGF2YQ=="
}
```



### Testing.

Payload.

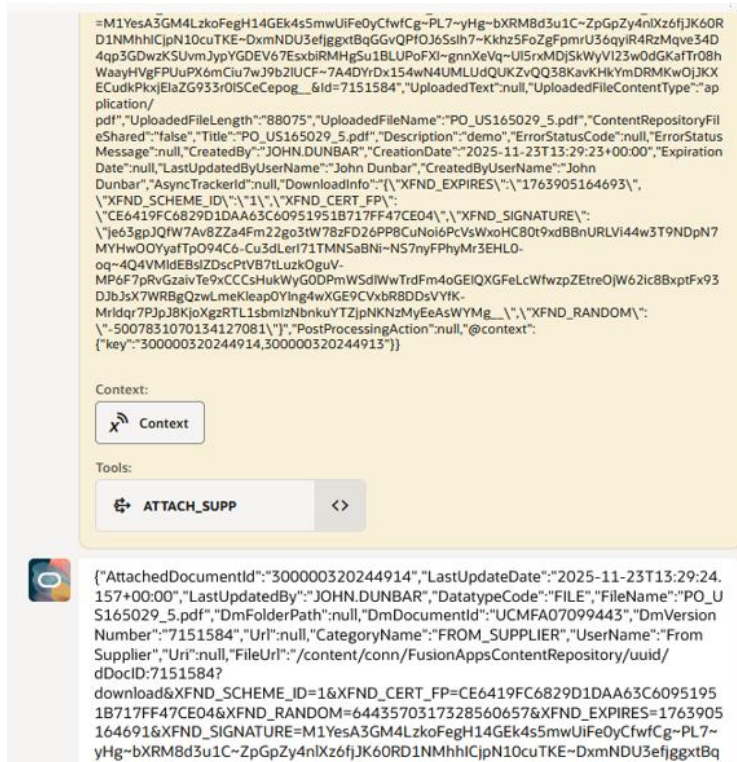
```
{
  "pDirectory": "/files/Nordea/AP_Invoices/Outbound/LOG",
  "pFileName": "PO_US165029_5.pdf",
  "pSupplier": "Lee Supplies"
}
```



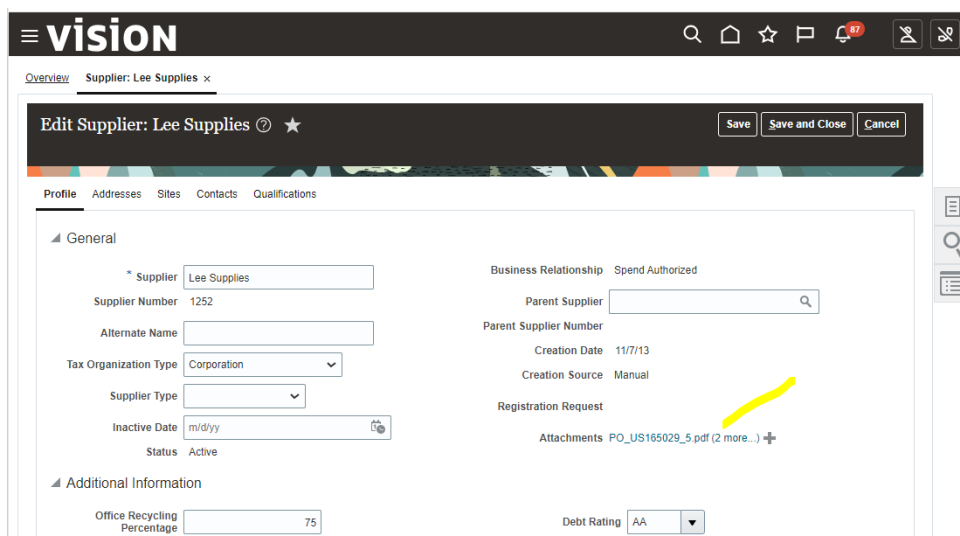
## Testing 2.

### Payload.

```
{
  "pDirectory": "/files/Nordea/AP_Invoices/Outbound/LOG",
  "pFileName": "PO_US165029_3.pdf",
  "pSupplier": "Lee Supplies"
}
```



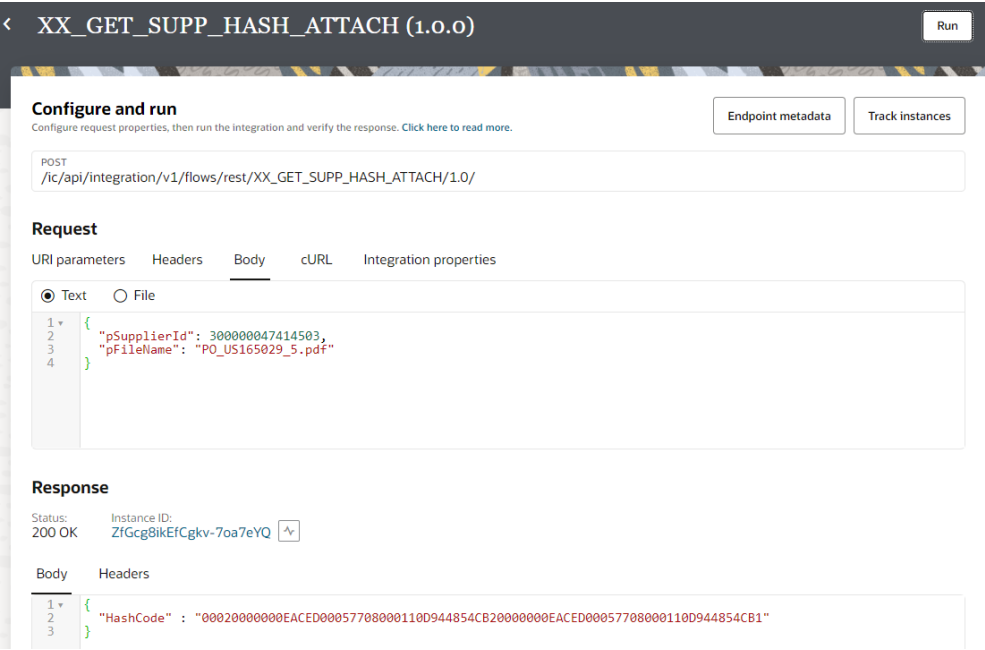
Notice in SaaS the attachment file.





In the next step we will get the hash code to get the attachment content using our OIC integration, the purpose of this step is to read the PDF content in base64 using document processor.

This OIC integration will be needed in further steps to retrieve the file content from attachment. The goal is to get the hash code needed.



We will need a new tool for REST External.

### **Get hash code. Tool of type REST External (6).**

Name: AA\_GET\_HASH\_FOR\_SUPP\_ATTACH.

Description: get hash for supplier attachment.

Note: needed to get the file content of attachment. There is no direct access to this value, and OIC integration has been developed and is provided.

### **Authorization.**

Instance URL:

<https://oic3-partners-bots4saas-yu.integration.ca-montreal-1.ocp.oraclecloud.com>

The screenshot shows the configuration page for a tool named 'AA\_GET\_HASH\_FOR\_SUPP\_ATTACH'. The page has a dark red header with the tool name and 'Cancel' and 'Create' buttons. Below the header, there are several input fields: 'Tool Type' (External REST), 'Tool Name' (AA\_GET\_HASH\_FOR\_SUPP\_ATTACH), 'Tool Code' (AAGETHASHFORSUPPATTACH), 'Family' (PRC), and 'Product' (Supplier). A description field contains the text 'get hash for supplier attachment'. Below these fields, there is a toggle for 'Require human approval' which is currently turned off. At the bottom, there is a tabbed interface with 'Authorization' and 'Functions' tabs. The 'Authorization' tab is active, showing the 'Instance URL' (https://oic3-partners-bots4saas-yu.integration.ca-montreal-1.ocp.oraclecloud.com) and 'Authentication' (Basic). There is also an 'Edit' button next to the 'Authentication' field.

### **Endpoint for Function.**

Name: get\_hash\_for\_supp\_attachment

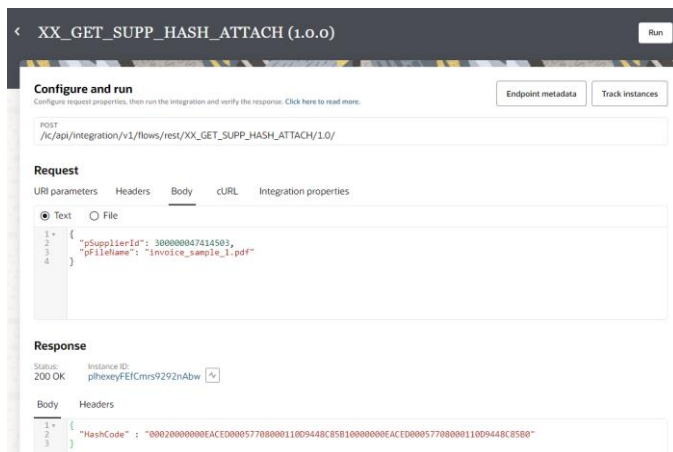
Operation: POST.

Description: get hash for supplier attachment.

Resource path:

`/ic/api/integration/v1/flows/rest/XX_GET_SUPP_HASH_ATTACH/1.0/`

OIC integration provided:



Description: get hash for supplier attachment.

**Add Endpoint**

Name:  Operation Type: HTTP POST

Resource Path:

Description:

Body Template: 

```
{
  "pSupplierId": {pSupplierId},
  "pFileName": "{pFileName}"
}
```

**Parameters** Sample Queries Headers

+ Add

Name: pFileName	Data Type: String
Description: pFileName	
Name: pSupplierId	Data Type: String
Description:	

Body template:

```
{
  "pSupplierId": {pSupplierId},
  "pFileName": "{pFileName}"
}
```

Require human approval


☐

Enable if you want human approval before running the tool.

Authorization

Functions

+ Add

Name	Operation Type	
get_hash_for_supp_attachment	POST	 

Sample queries.

```
{ "pSupplierId": 300000047414503, "pFileName": "PO_US165029_5.pdf" }
```

Parameters

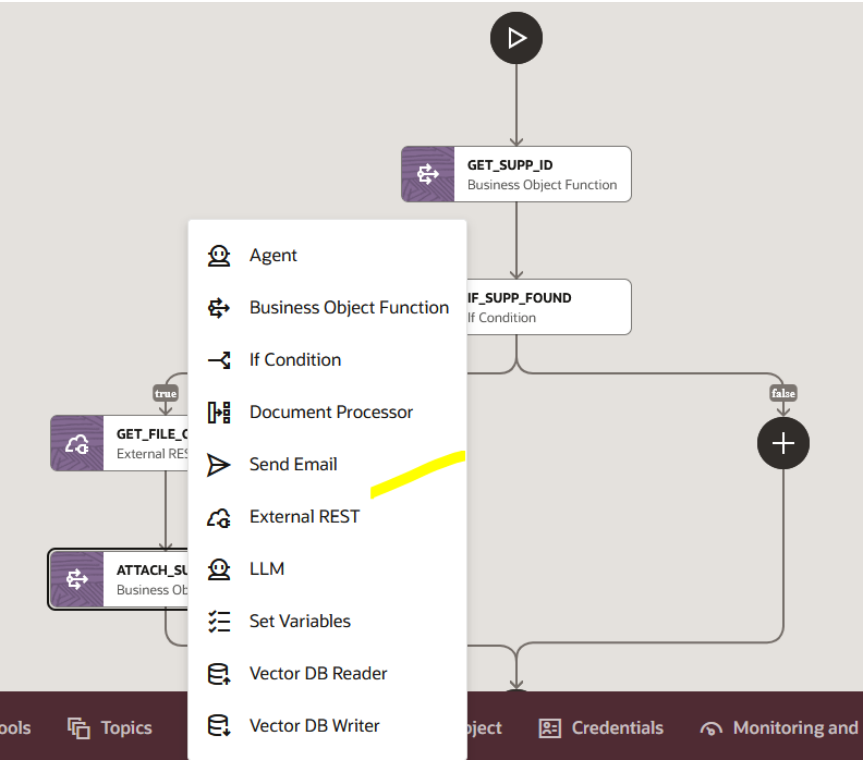
Sample Queries

Headers

+ Add

Query	Description
<pre>{   "pSupplierId": 300000047414503,   "pFileName": "PO_US165029_5.pdf" }</pre>	payload

Now we will add the node to our Agent Team.



Name in Agent team: AT\_GET\_HASH\_FOR\_SUPP\_ATTACH.  
Description: get hash for supplier attachment.  
Tool name: AT\_GET\_HASH\_FOR\_SUPP\_ATTACH.

**AT\_GET\_HASH\_FOR\_SUPP\_ATTACH**  
External REST

Name AT_GET_HASH_FOR_SUPP_ATTACH	Code ATGETHASHFORSUPPATTACH	Type External REST
Family FIN	Product Other	Tool Name AA_GET_HASH_FOR_SUPP_ATTACH
Function get_hash_for_supp_attachment		

**Parameters**

pFileName	string		
pSupplierId	string		

Parameter values.

### Parameters

Name	pFileName
Value	{{context.\$triggers.REST.\$input.pFileName}}
Description	

Output.

### Output

Output Builder Mode

☒ Specification Mode
 ☐ Simple Mode

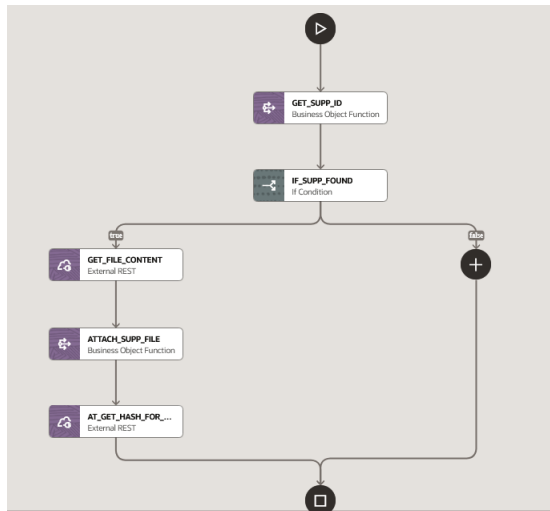
```

Properties
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "HashCode": {
      "type": "string"
    }
  }
}
  
```

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "properties": {
    "HashCode": {
      "type": "string"
    }
  }
}
  
```



Now we have this:



Testing.

Payload.


```
{
  "pDirectory": "/files/Nordea/AP_Invoices/Outbound/LOG",
  "pFileName": "PO_US165029_3.pdf",
  "pSupplier": "Lee Supplies"
}
```


**AT\_GET\_HASH\_FOR\_SUPP\_ATTACH**


Latency  
3.46 seconds

Input:  
{ "pFileName": "PO\_US165029\_3.pdf", "pSupplierId": "300000047414503" }

Output:  
{  
 "HashCode":  
 "00020000000EACED00057708000110D9449D0E64000000EACED00057708000110D9449D0E63"  
}

Context:  
 Context

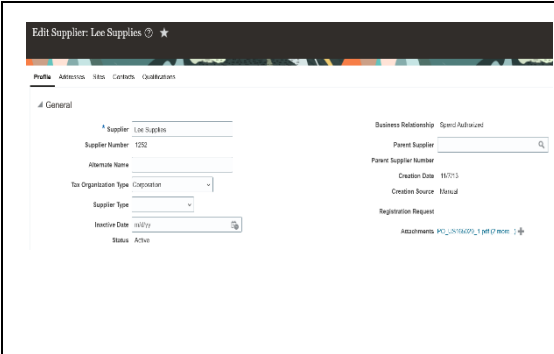
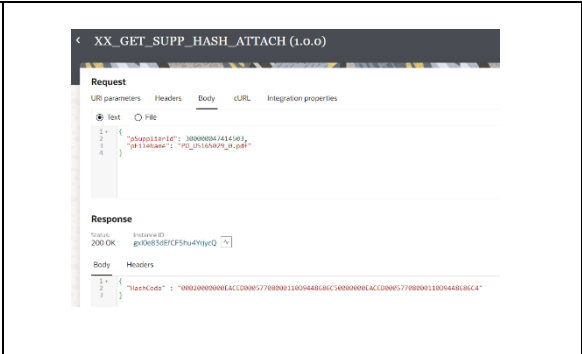
Tools:  
 AT\_GET\_HASH\_FOR\_SUPP\_ATTACH <>



```
{
  "HashCode":
    "00020000000EACED00057708000110D9449D0E64000000EACED00057708000110D9449D0E63"
}
```

Read file from attachment in supplier. Business Object (8).

Note: we are playing with this data for testing.

	
<p><b>Request.</b></p> <pre>{   "pSupplierId": 300000047414503,   "pFileName": "PO_US165029_0.pdf" }</pre>	<p><b>Response.</b></p> <pre>{   "HashCode" : "00020000000EACED00057708000110D9448686C50000000EACED00057708000110D9448686C4" }</pre>

Business Object Name: AA\_GET\_SUPPLIER\_ATTACHMENT.

Description: Retrieve the supplier attachment content.

Create a new business object

Family PRC	Product Supplier
Name AA_GET_SUPPLIER_ATTACHMENT	Code ORA_PRC_SUPPLIER_AAGETSUPPLIERATTACHMENT
Resource Type Monolith resource	Resource Path /fscmRestApi/resources/11.13.18.05/suppliers
Description Retrieve the supplier attachment content	

URL from Oracle reference doc.

/fscmRestApi/resources/11.13.18.05/suppliers/{supplierUniqID}/child/attachments/{attachmentsUniqID}

Sample URL to get the attachment content.

<https://fa-zzzz-saasfademo1.ds-fa.oraclepdemos.com:443/fscmRestApi/resources/11.13.18.05/suppliers/300000317626637/child/attachments/00020000000EACED00057708000110D9445D78CC0000000EACED00057708000110D9445D78CB/enclosure/FileContents>



Function-scenario.

Name: get\_supp\_attachment\_content

Description: get supplier attachment content.

Add a new scenario

Name

get\_supp\_attachment\_content

Description

get supplier attachment content

Operation Type

HTTP GET

Resource Path

/fscmRestApi/resources/11.13.18.05/suppliers/{pSupplierId}/child/attachments/{pAttachmentsUniqID}/enclosure/FileContents

Cancel

Done

Resource Path:

/fscmRestApi/resources/11.13.18.05/suppliers/{pSupplierId}/child/attachments/{pAttachmentsUniqID}/enclosure/FileContents

Parameters and example usage.

Parameters and Business Object Fields

Add Parameter

Add Field

Add Field from Specification

Name	Data Type	Description	Is Parameter	Actions
pSupplierId	string	Supplier id	true	<div><div></div><div></div></div>
pAttachmentsUniqID	string	Attachment Unique ID	true	<div><div></div><div></div></div>

Example Usages

+ Add

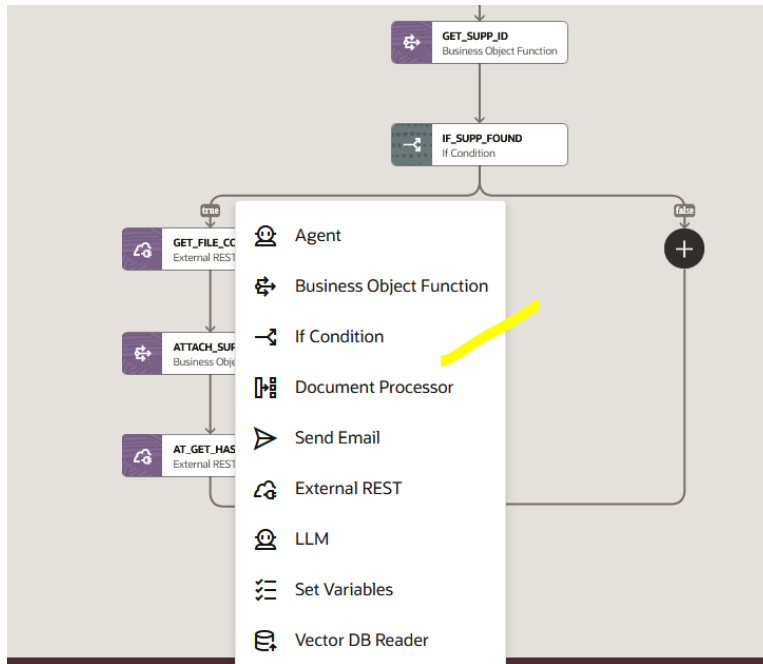
Description	Sample Payload	Actions
Payload	{ "pSupplierId": "0001", "pAttachmentsUniqID": "ABCD" }	<div><div></div><div></div></div>

Payload for example usage.

```
{
  "pSupplierId": "0001",
  "pAttachmentsUniqID": "0002000000EACED00057708000110D9448686C5000000EACED00057708000110D9448686C4"
}
```

### **Add “Document Processor” node using previous Business Object. (8).**

Use previous business object as basis for this section.



Name: ReadPDF.

Select Business Object: AA\_GET\_SUPPLIER\_ATTACHMENT.

**ReadPDF**  
Document Processor

Name ReadPDF	Code READPDF	Type Document Processor
Family PRC	Product Supplier	Business object AA_GET_SUPPLIER_ATTACHMENT
get_supp_attachment_content{		

Assign parameters.

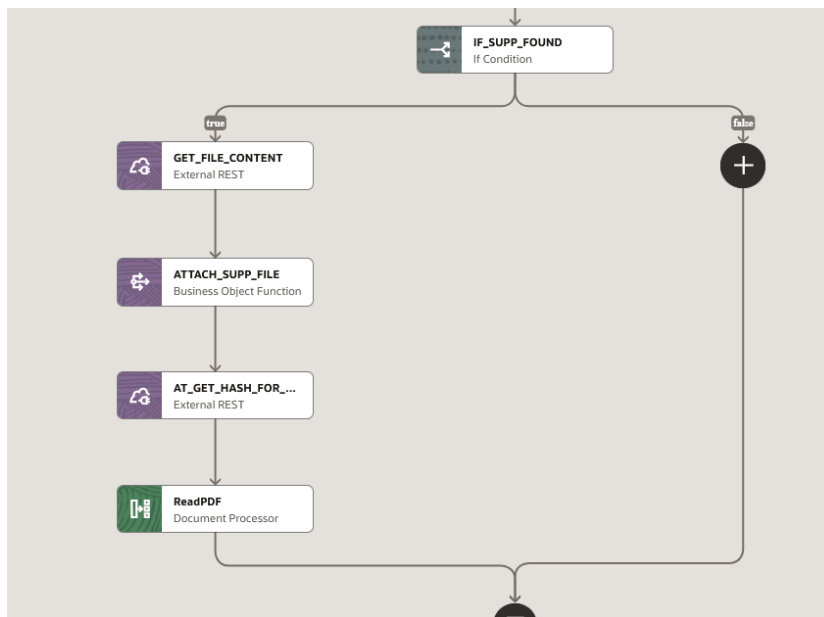
**Parameters**

Name pSupplierId	Type String
Value {{\$context.\$variables.varSupplierId}}	

Name	pAttachmentsUniqID
Value	{{context.\$nodes.ATGETHASHFORSUPPATTACH.\$output.HashCode}}
<i>Type {{ to start seeing context-sensitive expressions.</i>	

{{context.\$nodes.ATGETHASHFORSUPPATTACH.\$output.HashCode}}

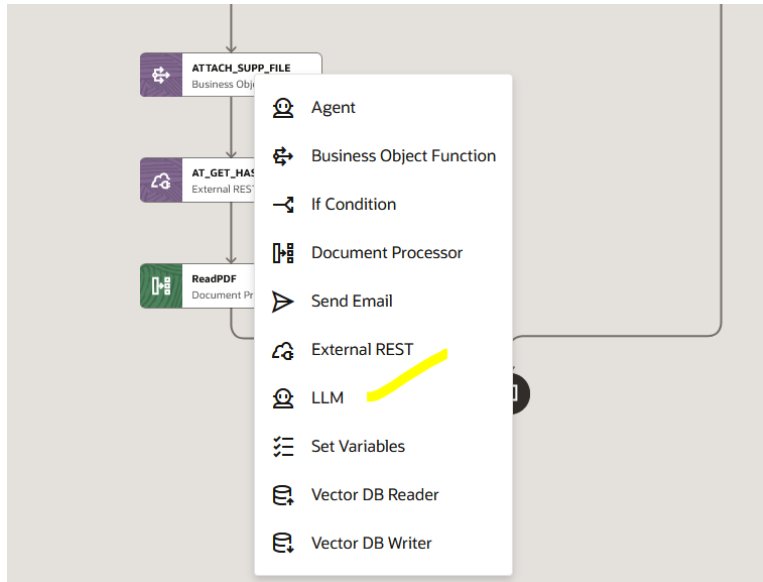
Now our canvas looks like this.



## Steps in Agent Team. AA CREATE SO FROM FTP PDF.

### First LLM (9).

Name: DEMO\_LLM\_1.



### Prompt text in LLM to create Sales Order adapted from template “Purchase Order to Sales Order Converter”.

Note: we have made these adaptations in characters as in 25D showed error.

- Parenthesis.
- Removed references that do not exist in the workflow.
- These quotes: `null`.
- The character “&”.

You are an **Order Extraction and Slot-Filling AI** with fuzzy matching and industrial PO know-how. Transform **any** Purchase Order PO text into a valid **Sales Order JSON** for Oracle Order Management Cloud **Sales Orders for Order Hub** create endpoint.

# OBJECTIVE

1. Ingest raw PO text any layout.
2. Fuzzy-extract standard PO attributes.
3. Populate a predefined JSON with "p\*" variables and line objects.
4. Defaults: missing dates → {{Date.now}} ISO 8601, others → null.
5. Build a compliant order payload "header + lines + charges" per Oracle REST model; if a field is NOT confidently mapped, add it to "unmappedAttributes".

```
<--BEGIN PURCHASE ORDER TEXT-->
{{$context.$nodes.READPDF.$output.text}}
<--END PURCHASE ORDER TEXT-->
```

# HINT USAGE

- Treat each hint as a **candidate** PO→SO mapping line.
- If "approvedFlag='Y'" and distance < 0.12, prefer the hint; else use your fuzzy logic.
- Anything still ambiguous → add to "unmappedAttributes" , cap poFragment2 = 60 chars.

# PRE-PROCESSING **MAKE THE TEXT MACHINE-FRIENDLY**

- Strip boilerplate/watermarks/ToC/terms **e.g.**, "Confidential","Internal", page banners, multi-page legal T\andCs\*.
- Normalize whitespace; join hard-wrapped lines; de-hyphenate split tokens **"FIT-\nTread-2950"** → **"FIT-Tread-2950"**.\*
- Canonicalize numbers/currency **strip** " , " , " \$ " , handle **"7,252.00"** , **"2900.00"**.\*

- Parse dates in formats such as "July 18th 2025", "18-July-25", "01-AUG-2025", "7/27/2025", "9/1/2025".
- Treat tables and free-form equally; infer columns by repeated patterns.

```
# INTELLIGENT MATCHING RULES *FUZZY EQUIVALENTS*

* **PO Number** ← "PO #", "P.O. No", "PO No", "Purchase Order", "Receipt", "Receipt No.", "Order
<num>", "Standard Purchase Order <num>".
* **Buyer** ← "Buyer", "Requisitioner", "Ordered By", "Contact", "Requester".
* **Bill To** ← "Bill To", "From", "Billed To", "Remit To".
* **Ship To** ← "Ship To", "Deliver To", "Delivery address", "Destination".
* **Quantity** ← "Qty", "QTY", "Amount" *when paired with UOM/price*.
* **Unit Price** ← "Unit Price", "Price Each", "List Price", "Unit".
* **Item/SKU** ← "Item #", "Part Number", "SKU", "Product Code".
* **Payment Terms** ← "Payment Terms", "Terms".
* **Ship Via / Carrier** ← "Ship Method", "Carrier", "Method".
* **Priority** ← "Shipment Priority", "Priority".
* **FreightTerms** ← "Freight", "Prepaid freight", "Postpaid freight".
* **FOBPoint** ← "Free on Board points".
* **Requested / Promised / Delivery Date** ← "Request Date", "Requested", "Promised", "Delivery
Date".
* **Guardrail:** * **BuyingPartyName is the customer, not the supplier**; never take supplier block as
the buyer.
* **Guardrail:** * **BuyingPartyName, if it looks like an address then it is not correct element.
Typically address from the first text block may be followed by BuyingPartyName **; never take address
as the BuyingPartyName.

# ADDRESS HEURISTICS

1. If a **Delivery address** block exists at header level, prefer it for **Ship To**.
2. Else use an explicit **Ship To** block.
3. If multiple logistics addresses *e.g., warehouse list, shipping instructions with many sites*,
**don't** treat the list as the ship-to; use the order's primary Delivery/Ship-To and record the rest
in "unmappedAttributes" with a note like "multipleCandidateShipToAddresses".
4. **Bill To** often equals the company header address; confirm via labels **Bill To**, "From",
company letterhead*.
5. City casing must be preserved exactly for "pShipToCity".
6. County may follow City in the address block.
7. If State is given in complete name then use as-is , never change to short abbreviation unless
stated. i.e. if "New Hampshire" is given as state then donot change to "NH" explicitly , use is as
"New Hampshire".

# SALES ORDER REST API CONFORMITY *STRUCTURE and KEY CHILD OBJECTS*

Target the **Sales Orders for Order Hub** create payload and children *lines, charges,
chargeComponents*. For payments/terms, use the Payments child when available. *See "Create Sales
Orders" + Lines/Charges + Charge Components + Payments.*

# ATTRIBUTE MAPPING

## Top-Level *Header*

"""
- pSourceTransactionNumber ← *PO Number OR Order Number OR Receipt* +{{Date.now}}, truncate ≤ 28
chars
- pSourceTransactionId ← same as pSourceTransactionNumber, truncate ≤ 28 chars
- pBusinessUnitName ← {{context.$variables.businessUnitName}}
- pBuyingPartyName ← Customer entity placing the order *Bill To/Buyer*
- pBuyingPartyContactName ← Buyer/Requisitioner/Contact
- pCurrentDate ← {{context.$system.$currentDate}} in ISO *if absent* , in ISO Format
only like "YYYY-MM-DD".
- pSourceTransactionSystem ← Source System if present, else default "GPR"
- pPartialShipAllowedFlag ← true , unless explicitly mentioned to false.
- pRequestedFulfillmentOrganizationName ← {{context.$variables.businessUnitName}}
- pRequestingBusinessUnitName ← {{context.$variables.businessUnitName}}
- pTransactionalCurrencyName ← Steps to find currency match:
    1. Look for Currency field and Currency symbol from the purchase order document input
    coming
    2. If currency is not explicitly defined, determine the country from the vendor/buyer
    address and use the local currency name of that country. else default "US Dollar"
- pTransactionalCurrencyCode ← Steps to find currency match:
    1. Look for Currency field and Currency symbol from the purchase order document input
    coming
    2. If currency is not explicitly defined, determine the country from the vendor/buyer
    address and use the local currency code of that country.; else default "USD"
- pCustomerPONumber ← Raw PO number from header
- pFreightTerms ← FreightTerms specified in PO Document explicitly *Unless Explicitly Mentioned ,
donot pass*
- pSalesPersonName ← Sales Person Name *Unless Explicitly Mentioned , donot pass* .
- pFOBPoint ← Free on Board points. Identifies the ownership of the item while it is in transit , if
specified , otherwise null.
- pShipping Instructions ← Delivery Instructions or Shipping Instructions mentioned in PO ,
otherwise null
"""

*Note:* Use "pTransactionalCurrencyCode" when you can identify a currency code/symbol; if only a name
is present **"US Dollar"*, convert to code "USD".
```

```

## billToCustomer[*]

From **Bill To** / **From** area:

"""
billToCustomer[*].PartyName      ← pBuyingPartyName
billToCustomer[*].AddressLine    ← pBillToAddressLine
billToCustomer[*].City           ← pBillToCity
billToCustomer[*].State          ← pBillToState
billToCustomer[*].PostalCode     ← pBillToPostalCode
billToCustomer[*].Country        ← Country *if present; else try to derive based on the address and
currency , if not deterministic then default to "US"*
billToCustomer[*].ContactName    ← pBillToContactName
billToCustomer[*].County         ← pCounty
billToCustomer[*].Province       ← pProvince
billToCustomer[*].FirstName      ← first token of contact
billToCustomer[*].LastName       ← remainder of contact
*omit County/Province if absent*
"""

## shipToCustomer[*]

From **Delivery address** or **Ship To**::

"""
shipToCustomer[*].PartyName      ← pShipToPartyName
shipToCustomer[*].ContactName    ← pShipToContactName
shipToCustomer[*].FirstName      ← first token
shipToCustomer[*].LastName       ← remainder
shipToCustomer[*].AddressLine    ← pShipToAddressLine
shipToCustomer[*].City           ← pShipToCity *preserve case*
shipToCustomer[*].State          ← pShipToState
shipToCustomer[*].County         ← pCounty
shipToCustomer[*].Province       ← pProvince
shipToCustomer[*].PostalCode     ← pShipToPostalCode
shipToCustomer[*].Country        ← Country *if present; else try to derive based on the address and
currency , if not deterministic then default to "US"*
*omit County/Province if absent*
"""

"""

# LINES EXTRACTION

Detect each PO line whether tabular or free-form, including wrapped descriptions:

* "ProductNumber" ← "Item #", "SKU", "Part Number", "Product Code"; fall back to leading alphanumeric
token *e.g., "FIT-Tread-2950", "PEL-1540"*.
* "OrderedQuantity" ← from "QTY", default "1".
* "OrderedUOM" ← from UOM column; default "Each".
* "RequestedShipDate" ← from "Requested", "Request Date", "Delivery Date", or "Promised"; default
*{{Date.now**}} in ISO Format only like "YYYY-MM-DD".
* "Unit Price" ← any unit-price field near the line.
* Ignore header "Subtotal/Tax/Total/Discount" for line extraction; use them only for anomaly cross-
check.

For each line create:

"""
lines[*].SourceTransactionLineId // increment 1..n , Never pass null , it is mandatory field
lines[*].SourceTransactionLineNumber // increment 1..n, Never pass null , it is mandatory field
lines[*].SourceTransactionScheduleId // increment 1..n, Never pass null , it is mandatory field
lines[*].SourceScheduleNumber      // increment 1..n, Never pass null , it is mandatory field
lines[*].RequestedFulfillmentOrganizationName ← {{ $context.$variables.businessUnitName }}
lines[*].ProductNumber
lines[*].OrderedQuantity
lines[*].OrderedUOM
lines[*].RequestedShipDate
lines[*].PaymentTerms
lines[*].ShippingInstructions
"""

# PRICING: ONE CHARGE WITH TWO COMPONENTS PER LINE

Create a single Price charge per line:

"""
lines[*].charges[0].SourceChargeId    = "C" + SourceTransactionLineId
lines[*].charges[0].ApplyTo           = "Price"
lines[*].charges[0].PriceType         = "One time"
lines[*].charges[0].ChargeType        = "Sale"
lines[*].charges[0].ChargeSubType     = "Price"
lines[*].charges[0].PrimaryFlag       = true
lines[*].charges[0].SequenceNumber    = 1
lines[*].charges[0].PricedQuantity    = OrderedQuantity
"""

Charge Components *use extracted "Unit Price"*:

```

```

"""
CC1:
    SourceChargeComponentId      = ChargeId + "-CC1"
    PriceElementCode             = "QP_LIST_PRICE"
    PriceElementUsageCode        = "LIST_PRICE"
    HeaderCurrencyUnitPrice      = UnitPrice
    HeaderCurrencyExtendedAmount = UnitPrice * OrderedQuantity
    SequenceNumber               = 1

CC2:
    SourceChargeComponentId      = ChargeId + "-CC2"
    PriceElementCode             = "QP_NET_PRICE"
    PriceElementUsageCode        = "NET_PRICE"
    HeaderCurrencyUnitPrice      = UnitPrice
    HeaderCurrencyExtendedAmount = UnitPrice * OrderedQuantity
    SequenceNumber               = 2
"""

*These align with the **Charges** and **Charge Components** child resources of Order Lines.*

# CROSS-CHECKS and EDGE CASES

• If header currency is present *e.g., "All prices in USD"*, set "TransactionalCurrencyCode="USD"".
• If sum*line extended* ≠ header "Subtotal/Total" by more than 1% or a rounding cent, add a single anomaly with a short note *don't block output*.
• If multiple candidate ship-to addresses are found *e.g., long warehouse lists*, pick the labeled **Delivery**/**Ship To** block and add others to "unmappedAttributes" as suggestions.
• If buyer contact appears only once *e.g., "Requisitioner James Bond"*, use it for both "pBuyingPartyContactName" and "billToCustomer[*].ContactName" unless a specific bill-to contact exists.
• If PO header includes both "Order <num>" and "PO # <num>", prefer the explicitly labeled PO; else most frequent/earliest header occurrence.
• Truncate "pSourceTransactionNumber" and "pSourceTransactionId" to ≤ 28 chars.

# ANOMALY CAPTURE *REQUIRED*

Collect anything not confidently mapped:

"""json
"unmappedAttributes": [
    {
        "poFragment": "<raw snippet or label>",
        "expectedAttr": "<pVariableName or targetField>",
        "suggestedValue": "<best guess or null>",
        "correctedValue": null,
        "approvedFlag": false
    }
]
"""

# OUTPUT FORMAT *RETURN **ONLY** THIS JSON*

"""json
{
    "orderData": {
        "pSourceTransactionNumber": "...",
        "pSourceTransactionId": "...",
        "pBusinessUnitName": "...",
        "pBuyingPartyName": "...",
        "pBuyingPartyContactName": "...",
        "pCurrentDate": "2025-08-16",
        "pCustomerPONumber": "...",
        "TransactionalCurrencyCode": "...",
        "PaymentTermsCode": "...",
        "FreightTerms": "...",
        "FOBPoint": "...",
        "ShippingInstructions": "...",
        "billToCustomer": [
            {
                "PartyName": "...",
                "AddressLine": "...",
                "City": "...",
                "State": "...",
                "PostalCode": "...",
                "Country": "US",
                "County": "...",
                "ContactName": "...",
                "FirstName": "...",
                "LastName": "..."
            }
        ],
        "shipToCustomer": [
            {
                "PartyName": "...",
                "AddressLine": "...",
                "City": "...",
                "State": "...",

```

```

        "PostalCode": "...",
        "Country": "US",
        "County": "...",
        "ContactName": "...",
        "FirstName": "...",
        "LastName": "..."
    }
},
"pLines": [
    {
        "SourceTransactionLineId": 1,
        "SourceTransactionLineNumber": 1,
        "SourceTransactionScheduleId": 1,
        "SourceScheduleNumber": 1,
        "RequestedFulfillmentOrganizationName": "Vision Operations",
        "ProductNumber": "...",
        "PaymentTerms": "...",
        "ShippingInstructions": "...",
        "OrderedQuantity": 10,
        "OrderedUOM": "Each",
        "RequestedShipDate": "2025-08-16",
        "charges": [
            {
                "SourceChargeId": "C1",
                "ApplyTo": "Price",
                "PriceType": "One time",
                "ChargeType": "Sale",
                "ChargeSubType": "Price",
                "PrimaryFlag": true,
                "SequenceNumber": 1,
                "PricedQuantity": 10,
                "chargeComponents": [
                    {
                        "SourceChargeComponentId": "C1-CC1",
                        "PriceElementCode": "QP_LIST_PRICE",
                        "PriceElementUsageCode": "LIST_PRICE",
                        "HeaderCurrencyUnitPrice": 100,
                        "HeaderCurrencyExtendedAmount": 1000,
                        "SequenceNumber": 1
                    },
                    {
                        "SourceChargeComponentId": "C1-CC2",
                        "PriceElementCode": "QP_NET_PRICE",
                        "PriceElementUsageCode": "NET_PRICE",
                        "HeaderCurrencyUnitPrice": 100,
                        "HeaderCurrencyExtendedAmount": 1000,
                        "SequenceNumber": 2
                    }
                ]
            }
        ]
    }
],
"unmappedAttributes": []
}
"""

```

**\*\*Only output the JSON above—no extra narration or markdown.\*\***

**\*\*In output the JSON above—for any underived value , DO NOT PASS null. Create JSON as "Province": "" instead of "Province": null\*\***

## Temperature.

0.0

## Output to create Sales Order adapted from template “Purchase Order to Sales Order Converter”.

```

{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "Generated schema for Root",
  "type": "object",
  "properties": {
    "orderData": {
      "type": "object",
      "properties": {
        "SourceTransactionNumber": {
          "type": "string"
        }
      }
    }
  }
}

```



```

"SourceTransactionId": {
  "type": "string"
},
"SourceTransactionSystem": {
  "type": "string"
},
"BusinessUnitName": {
  "type": "string"
},
"BuyingPartyName": {
  "type": "string"
},
"BuyingPartyContactName": {
  "type": "string"
},
"CurrentDate": {
  "type": "string"
},
"CustomerPONumber": {
  "type": "string"
},
"TransactionalCurrencyCode": {
  "type": "string"
},
"PaymentTermsCode": {
  "type": "string"
},
"FreightTerms": {
  "type": "string"
},
"FOBPoint": {
  "type": "string"
},
"Salesperson": {
  "type": "string"
},
"ShippingInstructions": {
  "type": "string"
},
"billToCustomer": {
  "type": "array",
  "items": {
    "type": "object",
    "properties": {
      "PartyName": {
        "type": "string"
      },
      "AddressLine": {
        "type": "string"
      },
      "City": {
        "type": "string"
      },
      "State": {
        "type": "string"
      },
      "PostalCode": {
        "type": "string"
      },
      "Country": {
        "type": "string"
      },
      "County": {
        "type": "string"
      },
      "Province": {
        "type": "string"
      },
      "ContactName": {
        "type": "string"
      },
      "FirstName": {
        "type": "string"
      },
      "LastName": {
        "type": "string"
      }
    }
  },
  "required": [
    "PartyName",
    "AddressLine",
    "City",
    "State",
    "PostalCode",
    "Country",
    "County",
    "Province",
    "ContactName",

```

```

        "FirstName",
        "LastName"
    ]
}
},
"shipToCustomer": {
    "type": "array",
    "items": {
        "type": "object",
        "properties": {
            "PartyName": {
                "type": "string"
            },
            "AddressLine": {
                "type": "string"
            },
            "City": {
                "type": "string"
            },
            "State": {
                "type": "string"
            },
            "PostalCode": {
                "type": "string"
            },
            "Country": {
                "type": "string"
            },
            "County": {
                "type": "string"
            },
            "Province": {
                "type": "string"
            },
            "ContactName": {
                "type": "string"
            },
            "FirstName": {
                "type": "string"
            },
            "LastName": {
                "type": "string"
            }
        }
    },
    "required": [
        "PartyName",
        "AddressLine",
        "City",
        "State",
        "PostalCode",
        "Country",
        "County",
        "Province",
        "ContactName",
        "FirstName",
        "LastName"
    ]
},
"lines": {
    "type": "array",
    "items": {
        "type": "object",
        "properties": {
            "SourceTransactionLineId": {
                "type": "number"
            },
            "SourceTransactionLineNumber": {
                "type": "number"
            },
            "SourceTransactionScheduleId": {
                "type": "number"
            },
            "SourceScheduleNumber": {
                "type": "number"
            },
            "RequestedFulfillmentOrganizationName": {
                "type": "string"
            },
            "ProductNumber": {
                "type": "string"
            },
            "PaymentTerms": {
                "type": "string"
            },
            "ShippingInstructions": {
                "type": "string"
            }
        }
    },

```

```

    "OrderedQuantity": {
      "type": "number"
    },
    "OrderedUOM": {
      "type": "string"
    },
    "RequestedShipDate": {
      "type": "string"
    },
    "charges": {
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "SourceChargeId": {
            "type": "string"
          },
          "ApplyTo": {
            "type": "string"
          },
          "PriceType": {
            "type": "string"
          },
          "ChargeType": {
            "type": "string"
          },
          "ChargeSubType": {
            "type": "string"
          },
          "PrimaryFlag": {
            "type": "boolean"
          },
          "SequenceNumber": {
            "type": "number"
          },
          "PricedQuantity": {
            "type": "number"
          },
          "chargeComponents": {
            "type": "array",
            "items": {
              "type": "object",
              "properties": {
                "SourceChargeComponentId": {
                  "type": "string"
                },
                "PriceElementCode": {
                  "type": "string"
                },
                "PriceElementUsageCode": {
                  "type": "string"
                },
                "HeaderCurrencyUnitPrice": {
                  "type": "number"
                },
                "HeaderCurrencyExtendedAmount": {
                  "type": "number"
                },
                "SequenceNumber": {
                  "type": "number"
                }
              }
            },
            "required": [
              "SourceChargeComponentId",
              "PriceElementCode",
              "PriceElementUsageCode",
              "HeaderCurrencyUnitPrice",
              "HeaderCurrencyExtendedAmount",
              "SequenceNumber"
            ]
          }
        }
      },
      "required": [
        "SourceChargeId",
        "ApplyTo",
        "PriceType",
        "ChargeType",
        "ChargeSubType",
        "PrimaryFlag",
        "SequenceNumber",
        "PricedQuantity",
        "chargeComponents"
      ]
    },
    "required": [

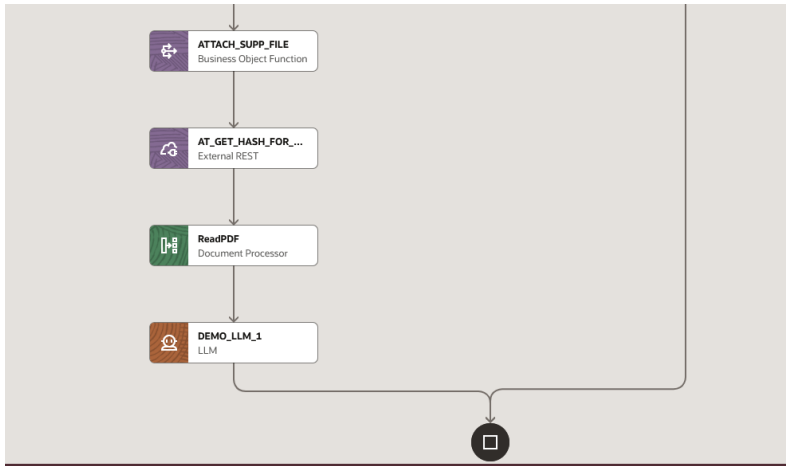
```

```

        "SourceTransactionLineId",
        "SourceTransactionLineNumber",
        "SourceTransactionScheduleId",
        "SourceScheduleNumber",
        "RequestedFulfillmentOrganizationName",
        "ProductNumber",
        "PaymentTerms",
        "ShippingInstructions",
        "OrderedQuantity",
        "OrderedUOM",
        "RequestedShipDate",
        "charges"
    ]
  }
}
},
"required": [
  "SourceTransactionNumber",
  "SourceTransactionId",
  "SourceTransactionSystem",
  "BusinessUnitName",
  "BuyingPartyName",
  "BuyingPartyContactName",
  "CurrentDate",
  "CustomerPONumber",
  "TransactionalCurrencyCode",
  "PaymentTermsCode",
  "FreightTerms",
  "FOBPoint",
  "Salesperson",
  "ShippingInstructions",
  "billToCustomer",
  "shipToCustomer",
  "lines"
]
},
"unmappedAttributes": {
  "type": "array",
  "items": {
    "type": "object",
    "properties": {
      "poFragment": {
        "type": "string"
      },
      "expectedAttr": {
        "type": "string"
      },
      "suggestedValue": {},
      "correctedValue": {},
      "approvedFlag": {
        "type": "boolean"
      }
    }
  },
  "required": [
    "poFragment",
    "expectedAttr",
    "suggestedValue",
    "correctedValue",
    "approvedFlag"
  ]
}
},
"required": [
  "orderData",
  "unmappedAttributes"
]
}

```

**Now the overview look like this.**



Testing.

Payload.

```
{
  "pDirectory": "/files/Nordea/AP_Invoices/Outbound/LOG",
  "pFileName": "PO_US165029_3.pdf",
  "pSupplier": "Lee Supplies"
}
```

These are (source) samples from PDF.

Purchase Order US165029

Order	US165029
Order Date	06-JAN-2025
Change Order	0
Change Order Date	06-JAN-2025
Revision	0
Ordered	1,017.00 USD

Sold To US1 Legal Entity  
500 ORACLE PKWY  
REDWOOD CITY, CA 94065 SAN MATEO

Supplier Lee Supplies  
5000 CARTER DR.  
LOS ANGELES, CA 90032 LOS ANGELES

Bill To US1 Business Unit  
411 University Street  
SEATTLE, WA 98101 King  
UNITED STATES

Ship To

Line	Item	Price	Quantity	UOM	Ordered	Taxable
1	Black Laser Printer Toner Cartridge	79.00		Ea		
Supplier Item CMC04413						
This line references Document (Blanket Purchase Agreement) 52236.						
	Requested		1	Ea	79.00	✓
	Promised					
	1/17/25					
	Requested					
	1/17/25					
Requested and Promised Dates correspond to the date of arrival at the Ship-to Location.						
	Line Total				79.00	
2	Inkjet Cyan Cartridge	39.00		Ea		
Supplier Item CMC04409						
This line references Document (Blanket Purchase Agreement) 52236.						
3	LaserJet Pro 400 M401n Laser Printer	899.00		Ea		
Supplier Item HPP965547						
Supplier Configuration ID 51209						
This line references Document (Blanket Purchase Agreement) 52232.						
	Requested		1	Ea	899.00	✓
	Promised					
	1/15/25					
	Requested					
	1/15/25					
Requested and Promised Dates correspond to the date of arrival at the Ship-to Location.						
	Line Total				899.00	
					Total	1,017.00

This is the resulting payload of our test to create a sales order

LLM Call(0)

```
{
  "orderData": {
    "SourceTransactionNumber": "US165029",
    "SourceTransactionId": "US165029",
    "SourceTransactionSystem": "GPR",
    "BusinessUnitName": "US1 Legal Entity",
    "BuyingPartyName": "US1 Business Unit",
    "BuyingPartyContactName": "Calvin Roth",
    "CurrentDate": "2025-01-06",
    "CustomerPONumber": "US165029",
    "TransactionalCurrencyCode": "USD",
    "PaymentTermsCode": "Net 30",
    "FreightTerms": "Buyer pays Origin freight",
    "FOBPoint": "Origin",
    "Salesperson": "",
    "ShippingInstructions": "",
    "billToCustomer": {
      "PartyName": "US1 Business Unit",
      "AddressLine": "411 University Street",
      "City": "SEATTLE",
      "State": "WA",
      "PostalCode": "98101",
      "Country": "US",
      "Country": "King",
      "Province": "",
      "ContactName": "Calvin Roth",
      "FirstName": "Calvin",
      "LastName": "Roth"
    }
  },
  "shipToCustomer": [
    {
      "PartyName": "US1 Business Unit",
      "AddressLine": "411 University Street",
      "City": "SEATTLE",
      "State": "WA",
      "PostalCode": "98101",
      "Country": "US",
      "Country": "King",
      "Province": "",
      "ContactName": "Calvin Roth",
      "FirstName": "Calvin",
      "LastName": "Roth"
    }
  ],
  "lines": [
    {
      "SourceTransactionLineId": 1,
      "SourceTransactionLineNumber": 1,
      "SourceTransactionScheduleId": 1,
      "SourceScheduleNumber": 1,
      "RequestedFulfillmentOrganizationName": "",
      "ProductNumber": "CMC04413",
      "PaymentTerms": "Net 30",
      "ShippingInstructions": "",
      "OrderedQuantity": 1,
      "OrderedUOM": "Each",
      "RequestedShipDate": "2025-01-17",
      "charges": [
        {
          "SourceChargeId": "C1",
          "ApplyTo": "Price",
          "PriceType": "One time",
          "ChargeType": "Sale",
          "ChargeSubType": "Price",
          "PrimaryFlag": true,
          "SequenceNumber": 1,
          "PricedQuantity": 1,
          "chargeComponents": [

```

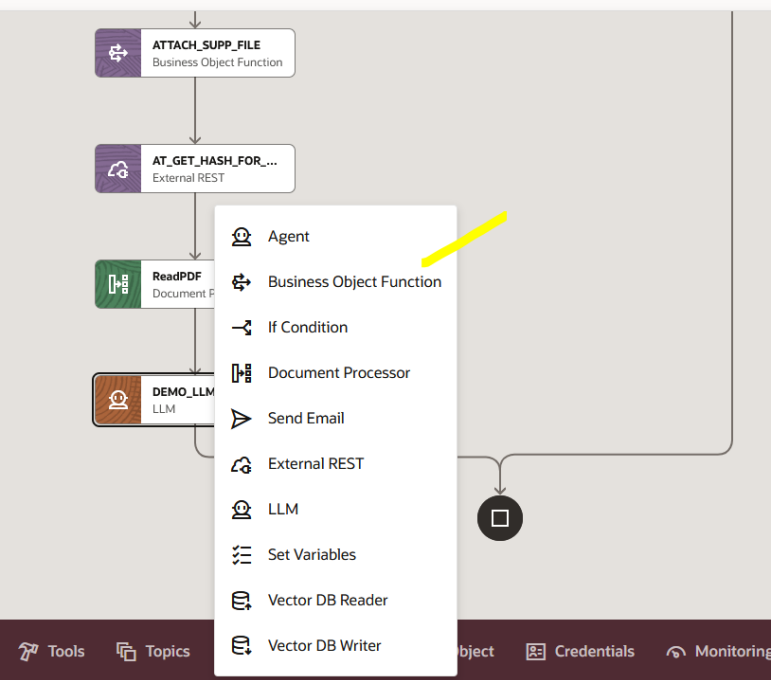
	<pre> "HeaderCurrencyExtendedAmount": 79, "SequenceNumber": 1 }, {   "SourceChargeComponentId": "C1-CC2",   "PriceElementCode": "QP_NET_PRICE",   "PriceElementUsageCode": "NET_PRICE",   "HeaderCurrencyUnitPrice": 79,   "HeaderCurrencyExtendedAmount": 79,   "SequenceNumber": 2 } ] }, {   "SourceTransactionLineId": 2,   "SourceTransactionLineNumber": 2,   "SourceTransactionScheduleId": 2,   "SourceScheduleNumber": 2,   "RequestedFulfillmentOrganizationName": "",   "ProductNumber": "CMC04409",   "PaymentTerms": "Net 30",   "ShippingInstructions": "",   "OrderedQuantity": 1,   "OrderedUOM": "Each",   "RequestedShipDate": "2025-01-17",   "charges": [     {       "SourceChargeId": "C2",       "ApplyTo": "Price",       "PriceType": "One time",       "ChargeType": "Sale",       "ChargeSubType": "Price",       "PrimaryFlag": true,       "SequenceNumber": 1,       "PricedQuantity": 1,       "chargeComponents": [         {           "SequenceNumber": 1         }       ]     }   ] }, {   "SourceChargeComponentId": "C2-CC2",   "PriceElementCode": "QP_NET_PRICE",   "PriceElementUsageCode": "NET_PRICE",   "HeaderCurrencyUnitPrice": 39,   "HeaderCurrencyExtendedAmount": 39,   "SequenceNumber": 2 } ] }, {   "SourceTransactionLineId": 3,   "SourceTransactionLineNumber": 3,   "SourceTransactionScheduleId": 3,   "SourceScheduleNumber": 3,   "RequestedFulfillmentOrganizationName": "",   "ProductNumber": "HPP9985847",   "PaymentTerms": "Net 30",   "ShippingInstructions": "",   "OrderedQuantity": 1,   "OrderedUOM": "Each",   "RequestedShipDate": "2025-01-15",   "charges": [     {       "SourceChargeId": "C3",       "ApplyTo": "Price",       "PriceType": "One time",       "ChargeType": "Sale",       "ChargeSubType": "Price",       "PrimaryFlag": true,       "SequenceNumber": 1,       "PricedQuantity": 1,       "chargeComponents": [         {           "SourceChargeComponentId": "C3-CC1",           "PriceElementCode": "QP_LIST_PRICE", </pre>
--	--

**Create Sales order. Business Object. (11).**

Duplicated from “Purchase Order to Sales Order Converter” template.

Business object				
<div>CreateOrdersForPOToSO <span>×</span> <input type="text" value="Search by business object name, ..."/></div> <div><span>Family</span> <span>Product</span></div> <div><span>+</span> Add</div>				
Name	Description	Seeded	Family	Product
CreateOrdersForPOToSO	This business object creates a new sales order with input for the lines and header level details.	Yes	SCM	Order Management
CreateOrdersForPOToSO_2	This business object creates a new sales order with input for the lines and header level details.	No	SCM	Order Management

**Include Business Object as Business Object Function.**





**CREATE\_SO**  
Business Object Function

Name:  Code:  Type:

Family:  Product:  Business object:

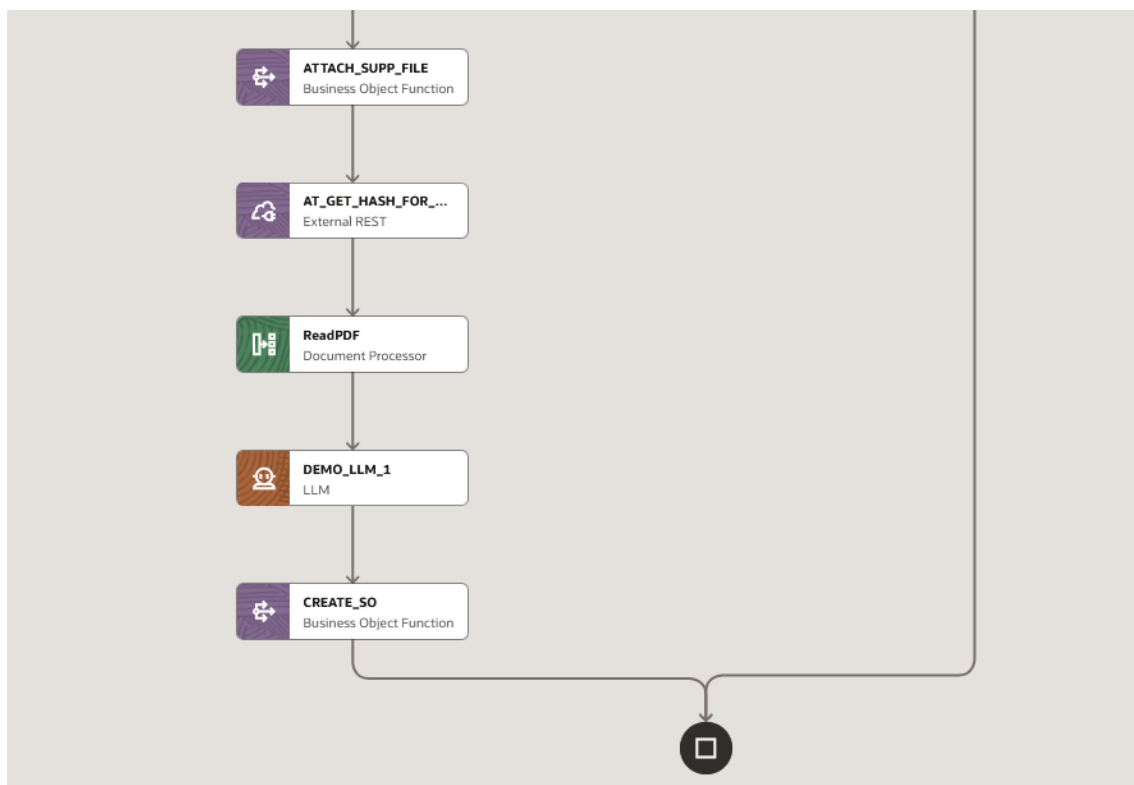
Function:

**Parameters**

pSourceTransactionNumber	string		
pSourceTransactionSystem	string		
pSourceTransactionId	string		
pBusinessUnitName	string		
pBuyingPartyName	string		

Name: CREATE\_SO

Now the overview looks like this:



### Mapping.

Only some values are mapped for learning purposes.

Out of scope to get a complete insert for now.

Parameters

Name

pSourceTransactionNumber

Type

String

Value

{{context.\$nodes.DEMOLLM1.\$output.orderData.SourceTransactionNumber}}

Name

pLines

Type

String

Value

{{context.\$nodes.DEMOLLM1.\$output.orderData.lines}}

## Testing.

### BOF\_CreateOrdersForPOToS0\_2

Latency  
0.13 seconds

Input:

```
{
  "pSourceTransactionSystem": "",
  "pSourceTransactionId": "",
  "pBusinessUnitName": "",
  "pBuyingPartyName": "",
  "pBuyingPartyContactName": "",
  "pCurrentDate": "",
  "pRequestedFulfillmentOrganizationName": "",
  "pRequestingBusinessUnitName": "",
  "pCustomerPONumber": "",
  "pCurrencyName": "",
  "pCurrencyCode": "",
  "pFreightTerms": "",
  "pSalesperson": "",
  "pBillToPartyName": "",
  "pBillToAddressLine": "",
  "pBillToCity": "",
  "pBillToState": "",
  "pBillToPostalCode": "",
  "pBillToCounty": "",
  "pBillToProvince": "",
  "pBillToCountry": "",
  "pBillToContactName": "",
  "pBillToFirstName": "",
  "pBillToLastName": "",
  "pShipToPartyName": "",
  "pShipToContactName": "",
  "pShipToFirstName": "",
  "pShipToLastName": "",
  "pShipToAddressLine": "",
  "pShipToCity": "",
  "pShipToState": "",
  "pShipToPostalCode": "",
  "pShipToCounty": "",
  "pShipToProvince": "",
  "pShipToCountry": "",
  "pSourceTransactionNumber": "US165029",
  "pLines": [
    {
      "OrderedUOM": "Ea",
      "charges": [
        {
          "SourceChargeId": "C1",
          "chargeComponents": [
            {
              "PriceElementCode": "QP_LIST_PRICE",
              "HeaderCurrencyExtendedAmount": 79,
              "SourceChargeComponentId": "C1-CC1",
              "PriceElementUsageCode": "LIST_PRICE",
              "HeaderCurrencyUnitPrice": 79,
              "SequenceNumber": 1
            },
            {
              "PriceElementCode": "QP_NET_PRICE",
              "HeaderCurrencyExtendedAmount": 79,
              "SourceChargeComponentId": "C1-CC2",
              "PriceElementUsageCode": "NET_PRICE",
              "HeaderCurrencyUnitPrice": 79,
              "SequenceNumber": 2
            }
          ]
        }
      ]
    }
  ]
}
```

```

    "pShipToCountry": "",
    "pSourceTransactionNumber": "US165029",
    "pLines": [
      {
        "OrderedUOM": "Ea",
        "charges": [
          {
            "SourceChargeId": "C1",
            "chargeComponents": [
              {
                "PriceElementCode": "QP_LIST_PRICE",
                "HeaderCurrencyExtendedAmount": 79,
                "SourceChargeComponentId": "C1-CC1",
                "PriceElementUsageCode": "LIST_PRICE",
                "HeaderCurrencyUnitPrice": 79,
                "SequenceNumber": 1
              },
              {
                "PriceElementCode": "QP_NET_PRICE",
                "HeaderCurrencyExtendedAmount": 79,
                "SourceChargeComponentId": "C1-CC2",
                "PriceElementUsageCode": "NET_PRICE",
                "HeaderCurrencyUnitPrice": 79,
                "SequenceNumber": 2
              }
            ],
            "PrimaryFlag": true,
            "ChargeType": "Sale",
            "ApplyTo": "Price",
            "SequenceNumber": 1,
            "PricedQuantity": 1,
            "ChargeSubType": "Price",
            "PriceType": "One time"
          }
        ],
        "SourceTransactionLineNumber": 1,
        "OrderedQuantity": 1,
        "ShippingInstructions": "",
        "RequestedShipDate": "2025-01-17",
        "SourceTransactionScheduleId": 1,
        "RequestedFulfillmentOrganizationName": "",
        "ProductNumber": "CMC04413",
        "PaymentTerms": "Net 30"
      }
    ]
  }
}

```

As we have mapped only a couple of nodes (source\_transaction\_number and lines) we get an incomplete payload, but the process is run.

Feel free to complete the mapping to get your sales order created.

## **Conclusion.**

We are able to read a PDF file, attach it to a supplier, and then read it and generate payload using Document Processor and LLM that could be used to create sales orders, purchase requisitions or another object changing the Prompt and Output.

## Technical

Code example	Comments
XX_GET_SUPP_HASH_ATTACH_01.00.0000.iar	Integration to be used as reference.
XX_AI_FF_GET_FILE_CONTE_01.00.0002.iar	Integration to be used as reference.
PO_US165029_0.pdf	PDF samples for testing.
PO_US165029_1.pdf	
...	
PO_US165029_5.pdf	