# Hands On Lab
# INFOCUS 2025

## AP Voucher Automation from Start to Finish

Trainee Guide

Tuesday, September 9

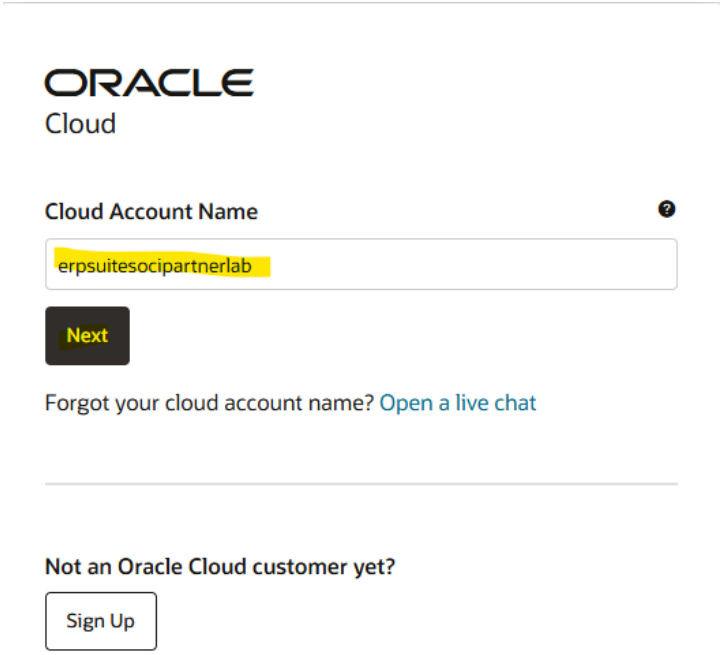
# Trainee Guide

**Log into ERP Suites Website to download the files**

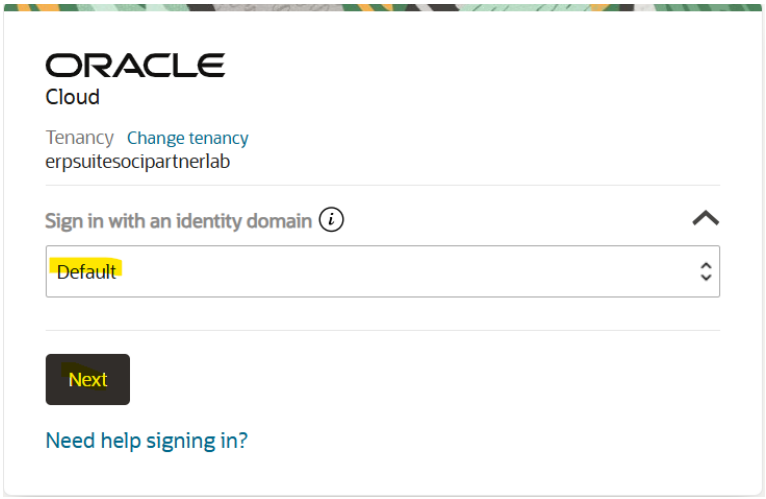
1. Go to erpsuites.com/lab to download AP Invoices for the lab
2. Once there click on **Download PDFs**


**Sign into the Console that Use Identity Domains**

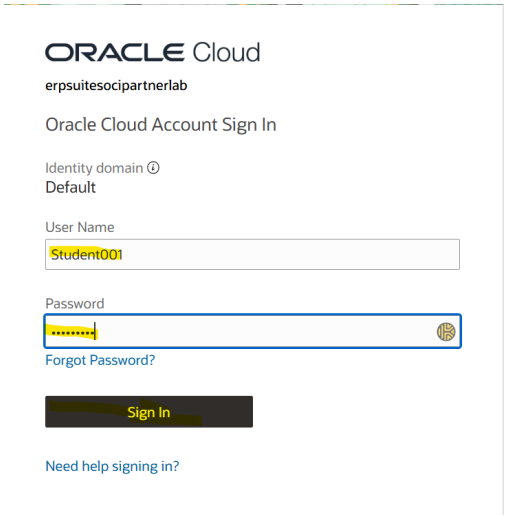
1. Go to http://cloud.oracle.com
2. Enter your tenancy name **erpsuitesocipartnerlab** under Cloud Account Name and click **Next**



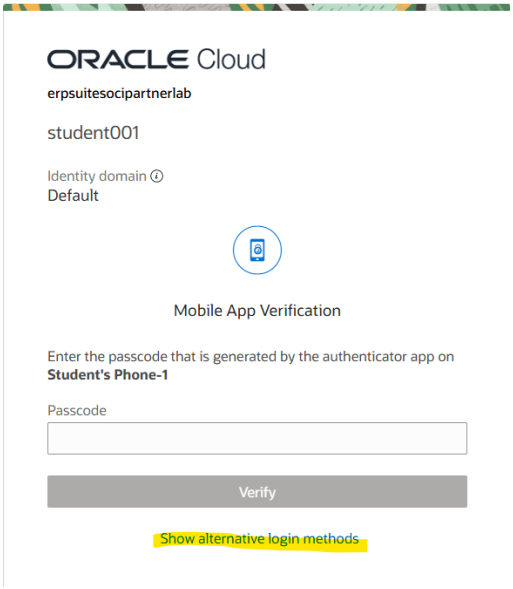

3. Sign in with the **Default** identity domain and click **Next**



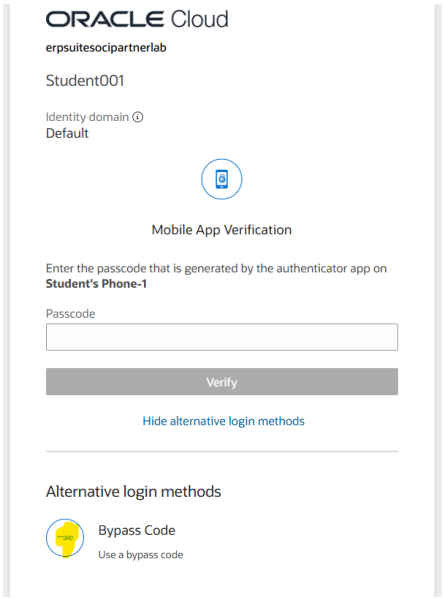

4. **Sign in with the credentials provided on the sheet in front of you**
    a. Entering username:
    b. Entering password
    c. Click **Sign In**



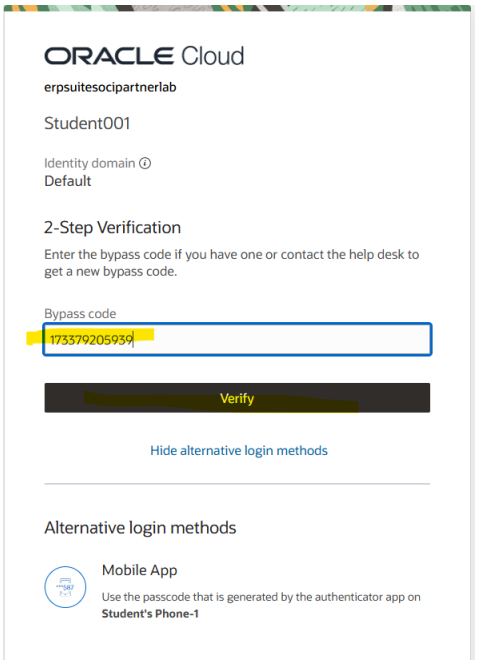
5. Next click on **Show Alternative login methods**




6. Next Select the **circle next to** bycass code



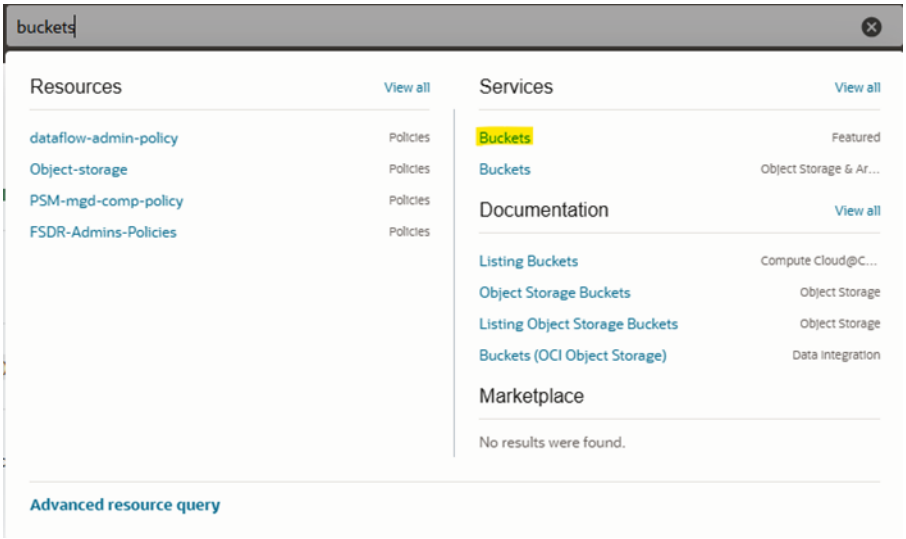
7. **Sign in with the credentials provided on the sheet in front of you** and then click **Verify**



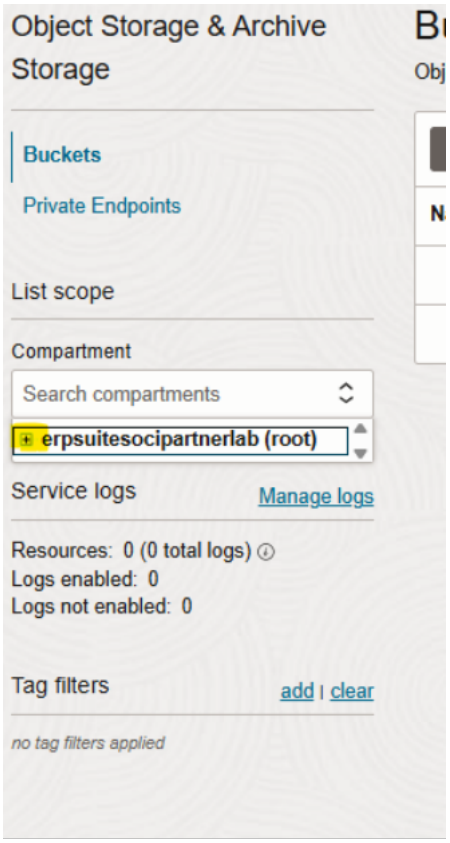

You are now signed into OCI

**Drop AP Invoices into an OCI Bucket**

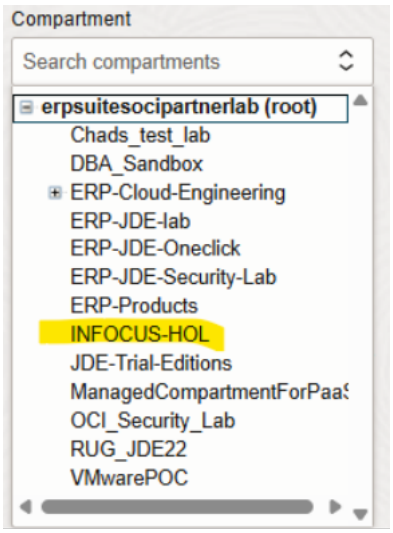
1. From the homepage navigate to the search bar above and type **Buckets** select the first option

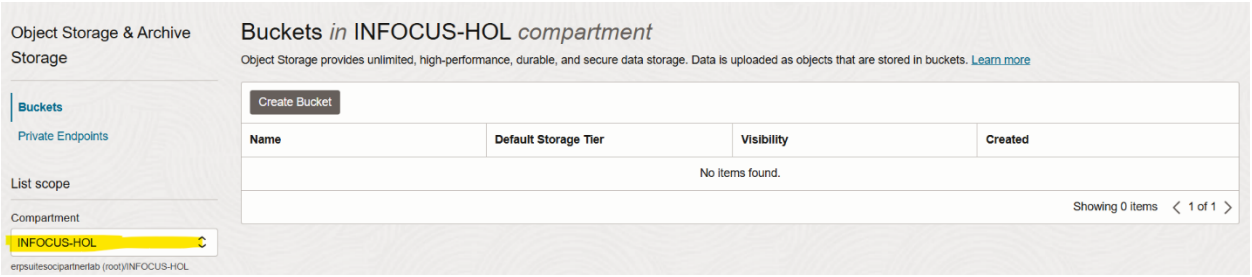


2. On the Left-hand side under choose the INFOCUS-HOL compartment
   a. Click the **+** sign to extend out all available compartments



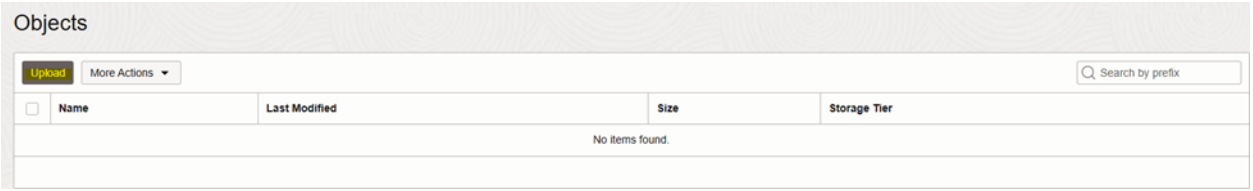

   b. Click on the **INFOCUS-HOL** compartment from the list





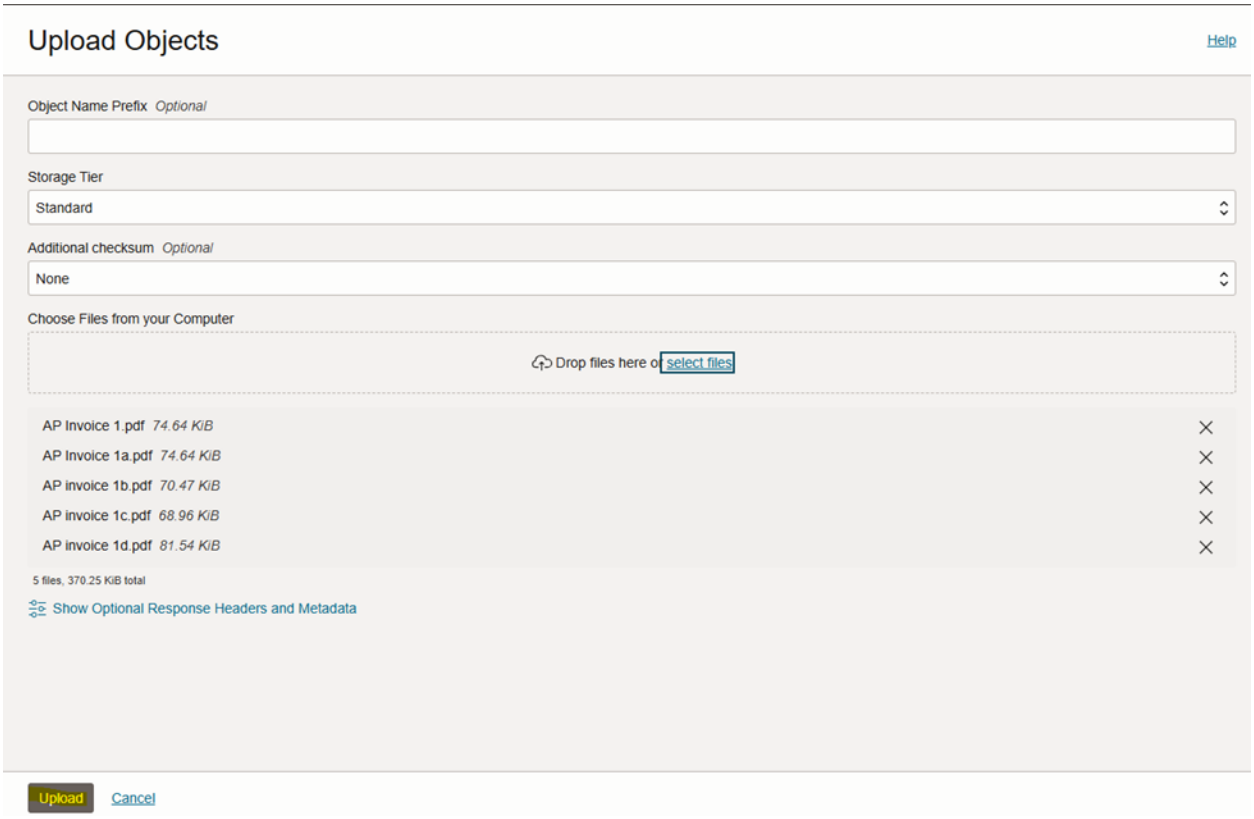

3. Next click on your assigned bucket number **Studentxxx**


4. Once inside your bucket click **Upload**




5. **Drag and Drop** files that you downloaded in the first step to the selected bucket.

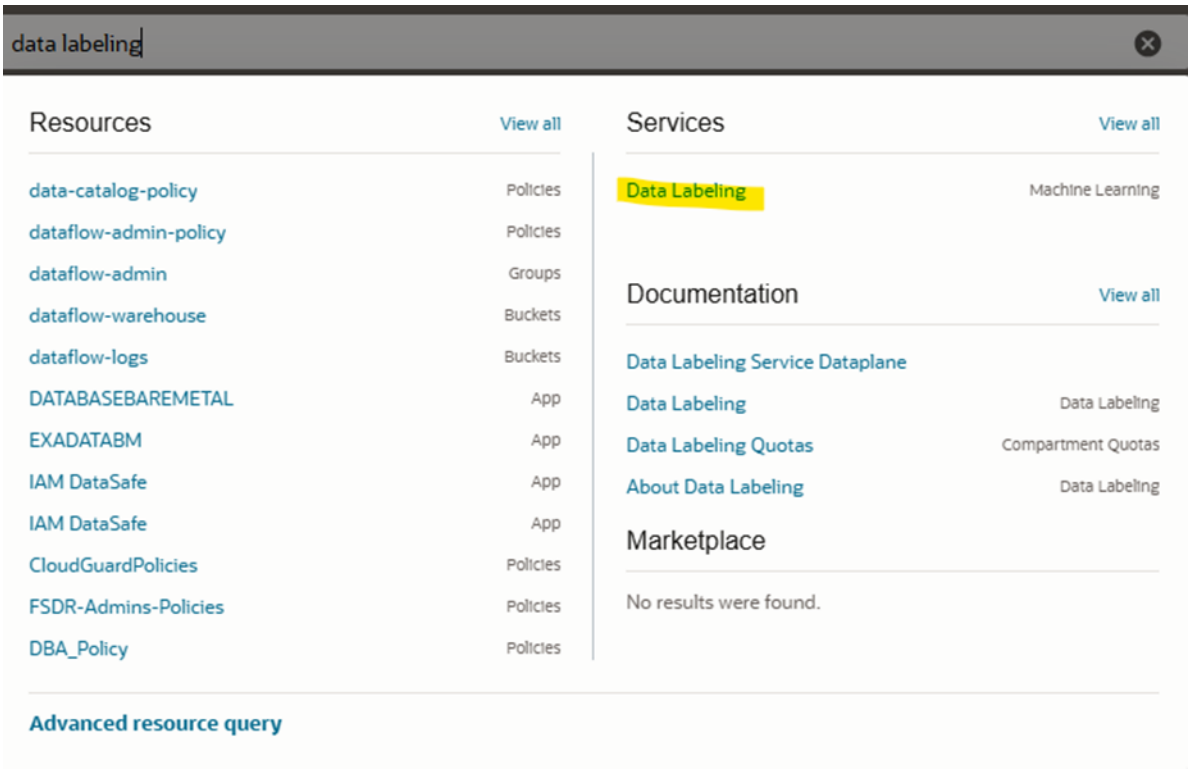
6. Once all the files appear on the screen click **Upload**



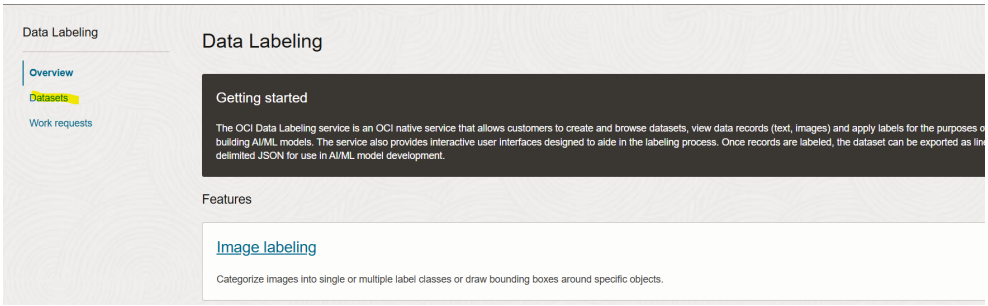

7. The files have been uploaded click **Close**

## Create a Dataset for AP Invoices

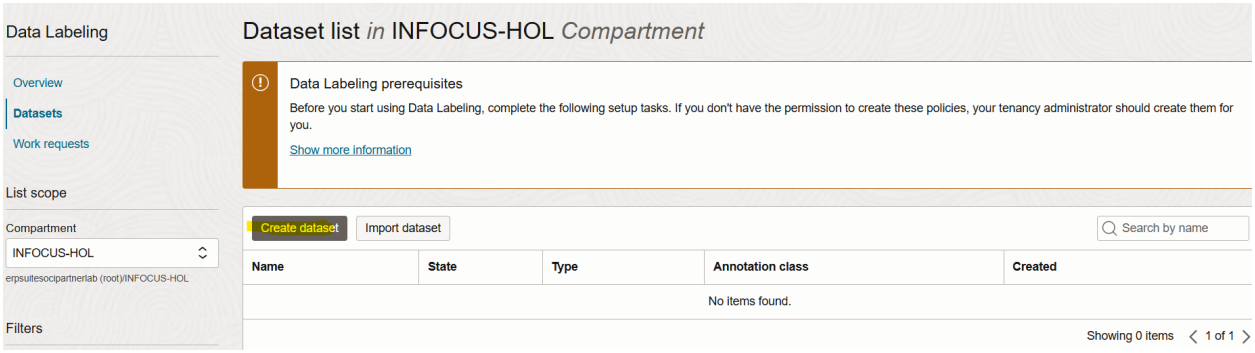
1. From the buckets page navigate to the search bar above and type **Data Labeling** and select the first option



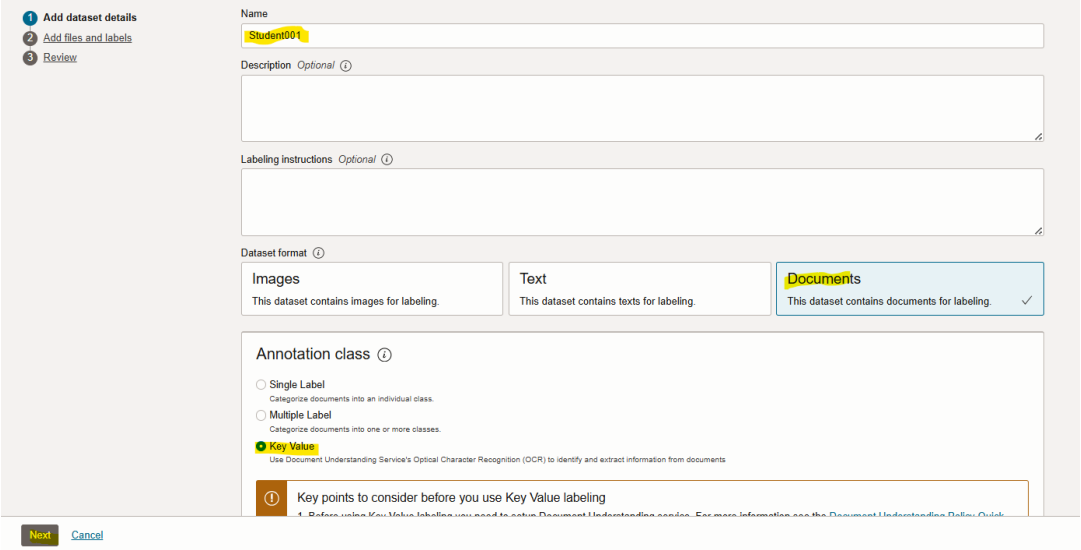

2. On the left-hand side of the Data Labeling page click **Datasets**



3. Next click **Create dataset**




a. Under Add dataset details
   i. Type name **Studentxxx**
   ii. Select dataset format **Documents**
   iii. Under annotation class select **Key Value**
   iv. Click **Next** in the bottom left-hand corner

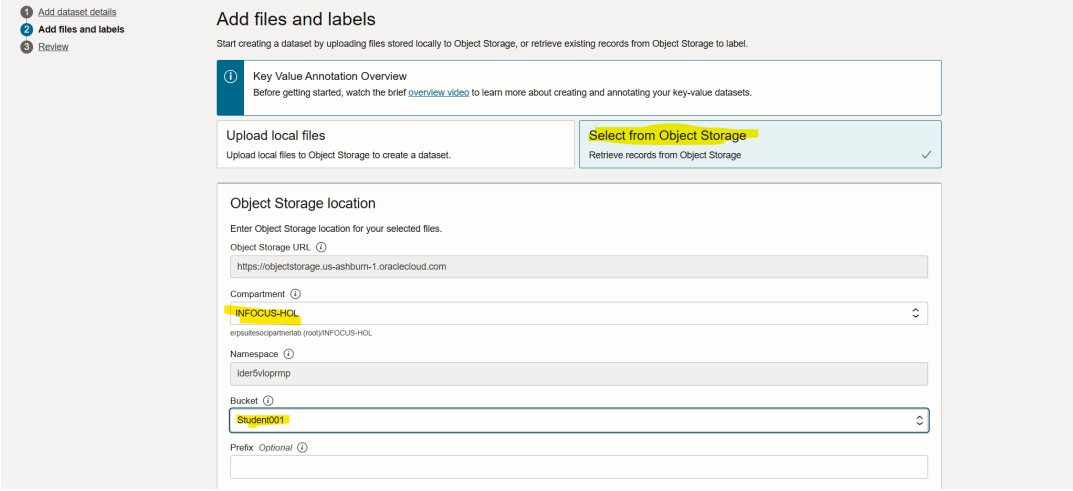


**b.** Under Add files and labels

i. Click **Select from Object Storage**

ii. Select Compartment **INFOCUS-HOL**

iii. Select the bucket you have created **Studentxxx**




The files you uploaded into the buckets will display below

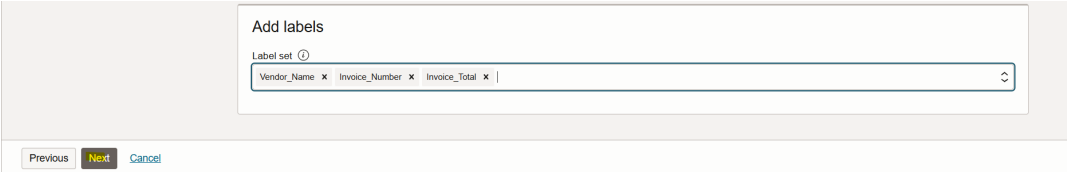
**1.** Scroll Down to **Add labels** to the text box.

**a.** Type **Vendor_Name** and press **Enter**

b. Type **Invoice_Number** and press **Enter**

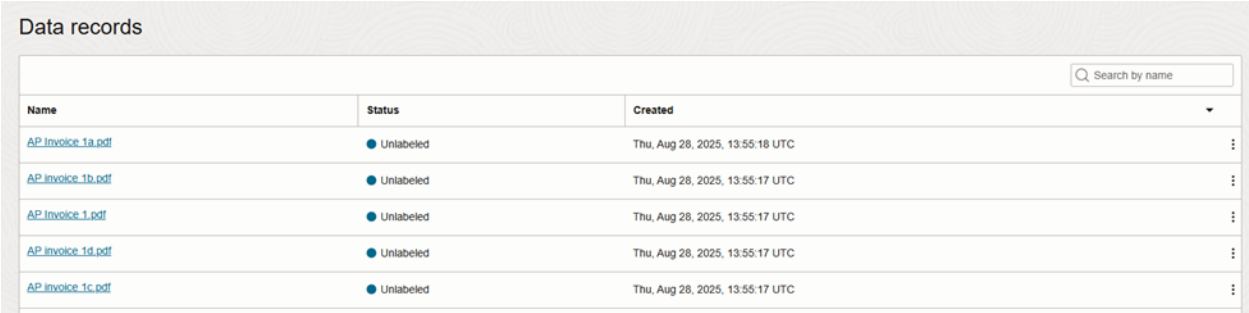
c. Type **Invoice_Total** and press **Enter**

iv. Click **Next** to review your dataset



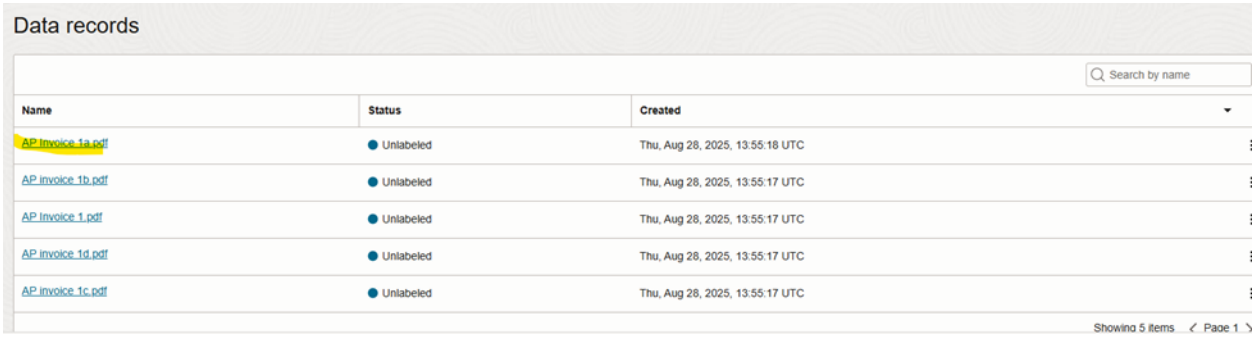

c. Under Review Click **Create** to create the dataset

You will see that the data records will start to get generated and show up as status **Unlabeled**

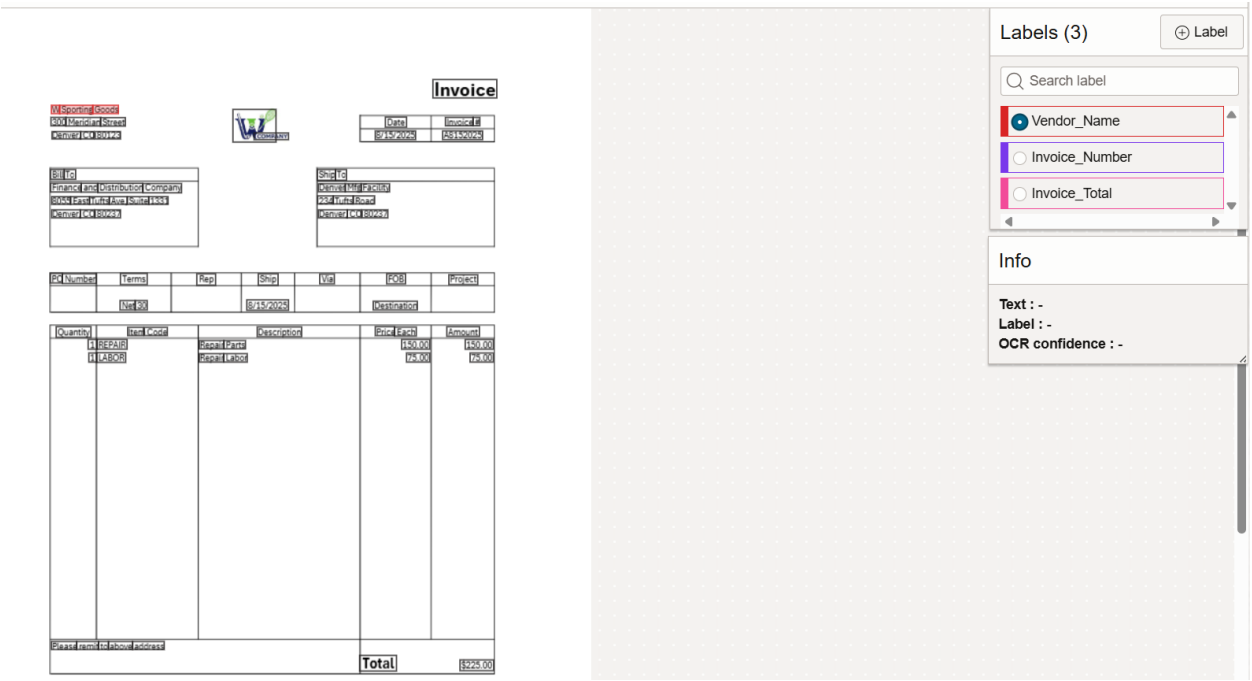



## Assigning Labels to AP Invoices

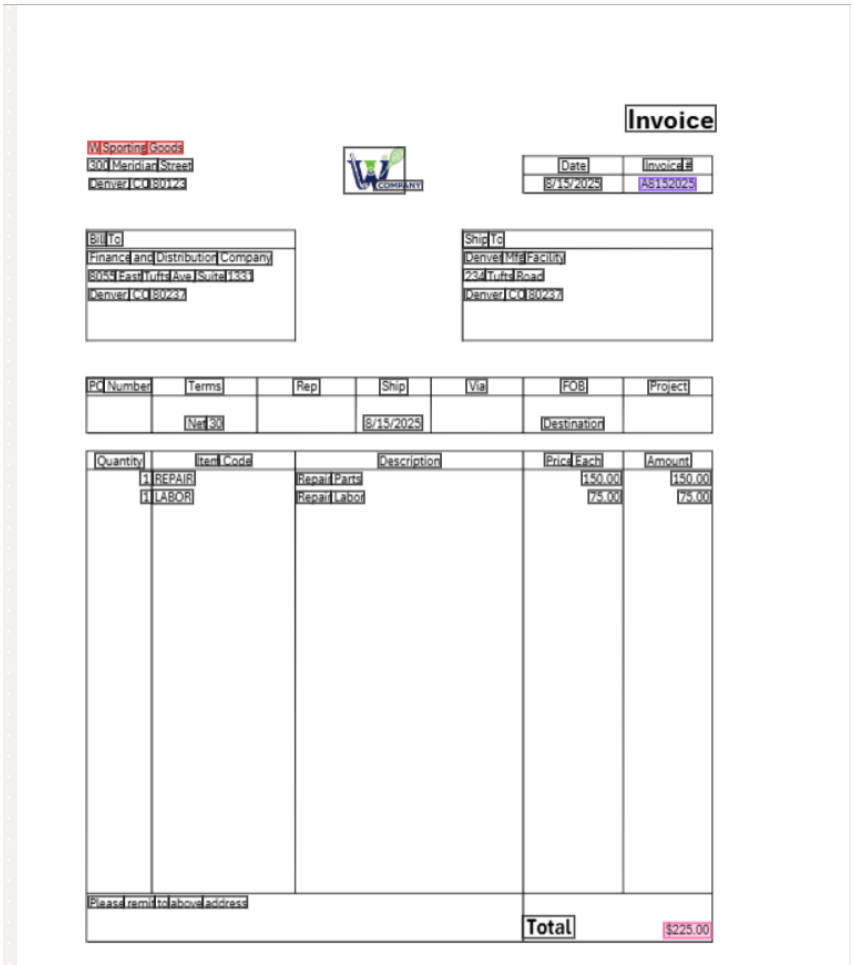
1. Click on the first **AP Invoice** to start labeling.



2. With your mouse **Drag a box** around the **Vendor_Name** of the PDF
3. Then select the **Vendor_Name** under labels on the right-hand side

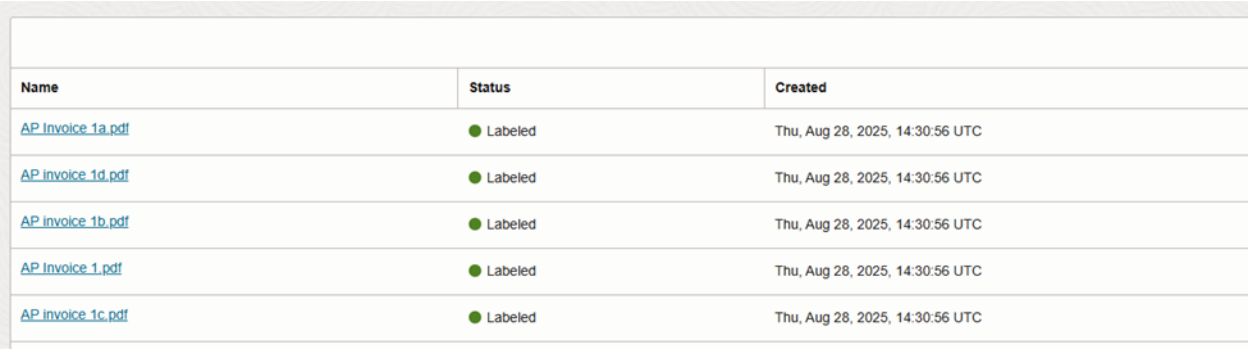


4. Repeat this step for all other AP Invoice fields. Below is a screenshot of where you should be labeling the values
    1. **Invoice_Number**
    2. **Invoice_Total**




5. Click **Save** & **Next**
6. Repeat the steps above for all 5 AP Invoices
7. On the last file Click **Save & Continue**

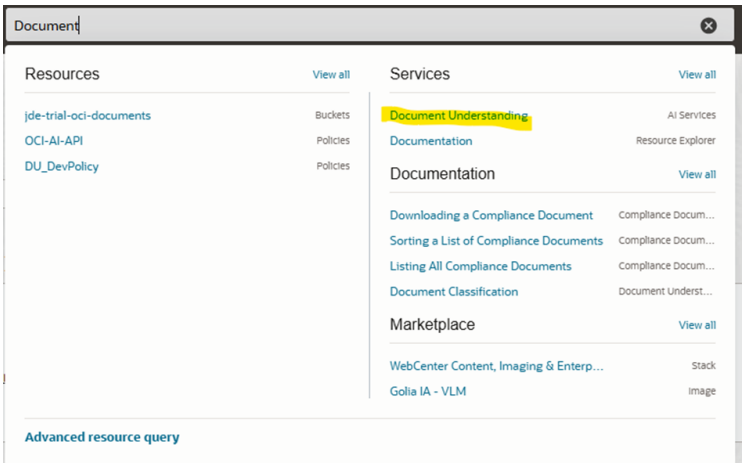
The dataset is now labeled and ready for the document understanding model



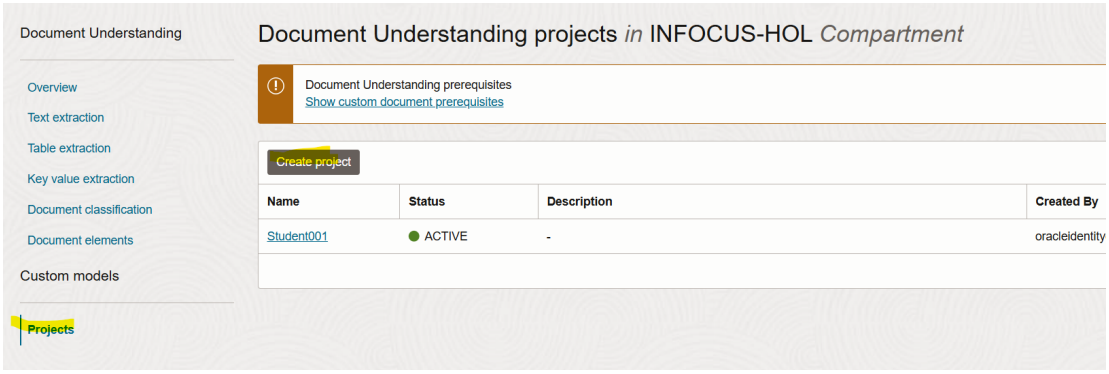
| Name | Status | Created |
|---|---|---|
| AP Invoice 1a.pdf | Labeled | Thu, Aug 28, 2025, 14:30:56 UTC |
| AP invoice 1d.pdf | Labeled | Thu, Aug 28, 2025, 14:30:56 UTC |
| AP invoice 1b.pdf | Labeled | Thu, Aug 28, 2025, 14:30:56 UTC |
| AP Invoice 1.pdf | Labeled | Thu, Aug 28, 2025, 14:30:56 UTC |
| AP invoice 1c.pdf | Labeled | Thu, Aug 28, 2025, 14:30:56 UTC |

## Create a Project

1. From the data labeling page navigate to the search bar above and type **Document Understanding** and click the first option

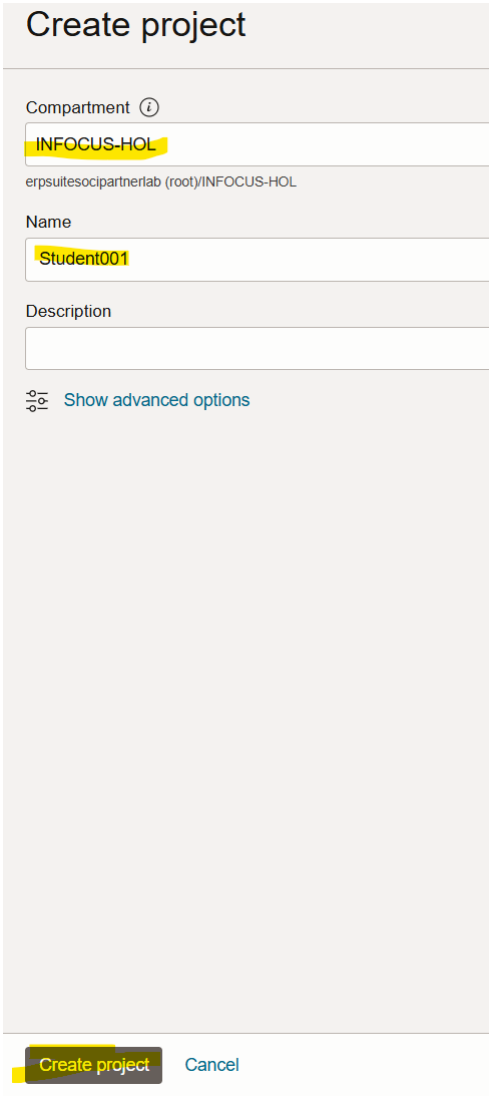



2. Under custom models on the left-hand side click **Projects**
3. Then click **Create Project**




   a. Select the compartment **INFOCUS-HOL**
   **b.** Enter a Project name **Studentxxx**

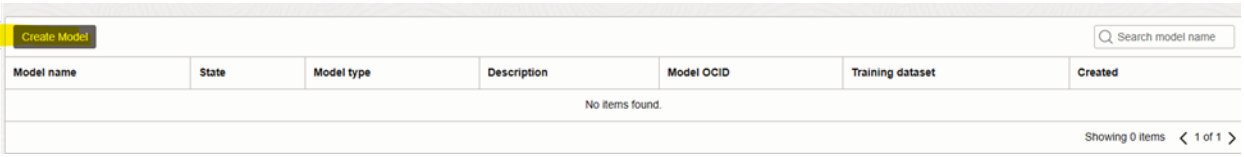
c. Click on the **Create Project**



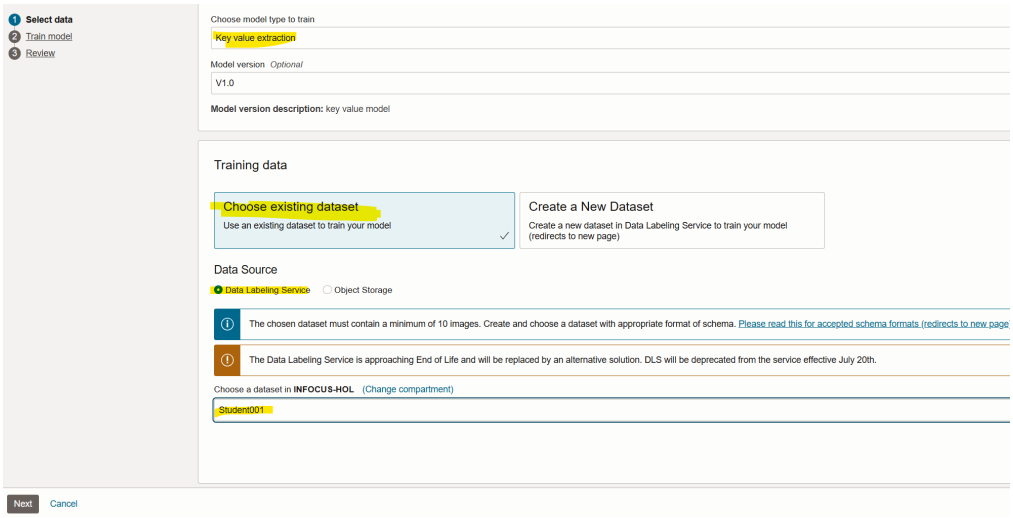

## Create Models to Train AP Invoices

1. Click on the Project you just created **Studentxxx**

2. Once the project is created click on **Create Model**




   a. Under Select Data
      i. Select **Key value extraction**
      ii. Click **Choosing existing dataset**
      iii. Under Data Source click **Data Labeling Service**
      iv. From the drop down choose **Studentxxx** created from data labeling
      v. Click **Next**



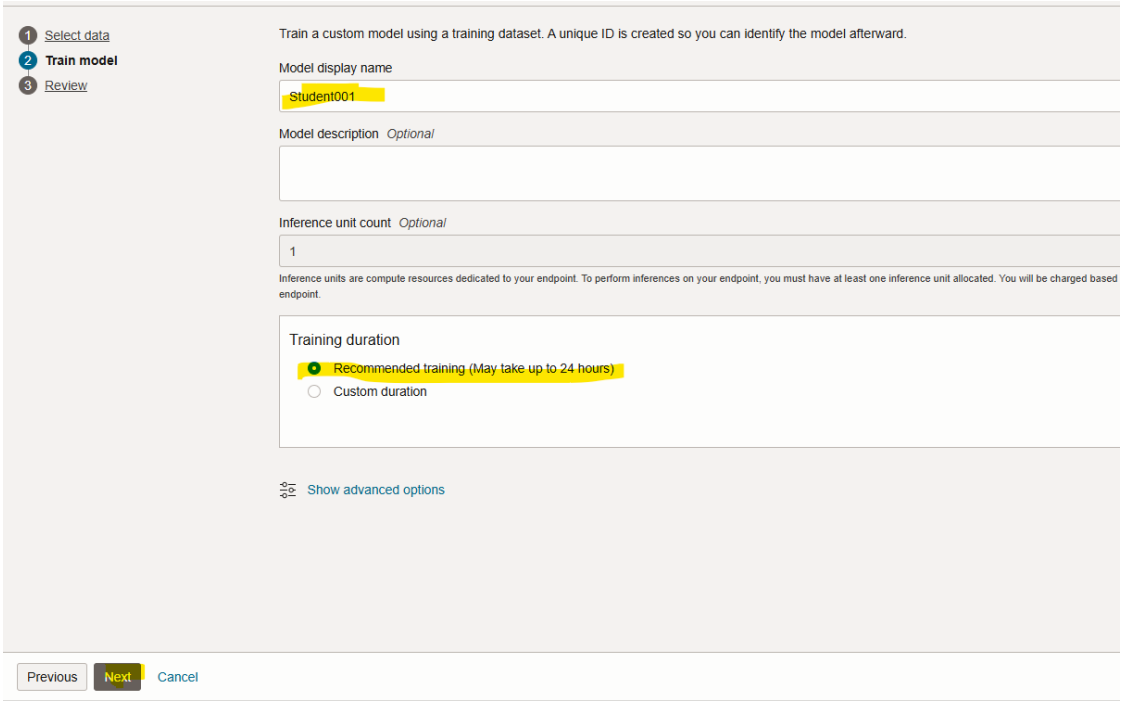
**b.** Under Train **model**

i. Create Model display name **Studentxxx**

ii. Select training language **English**

**iii.** Select training duration **Recommended training (May take up to 24 hours)**

**iv.** Click **Next**

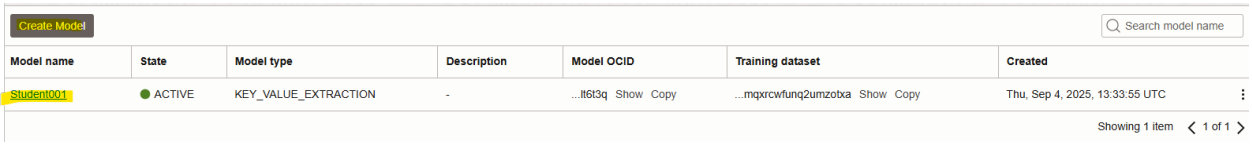



c. Click on **Review** to review all inputs

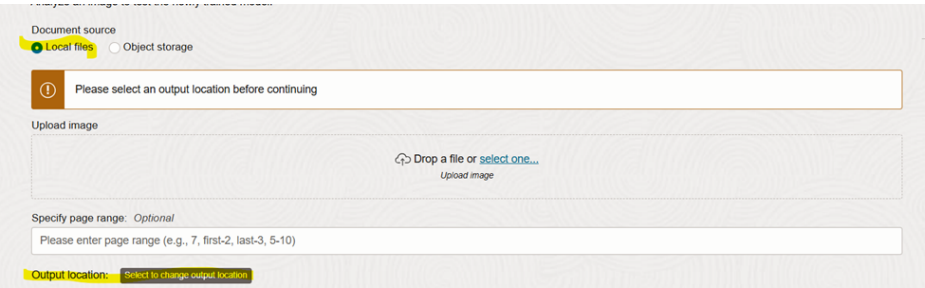
**3.** Click **Create and Train** to train your model

## Analyze the AP Invoices

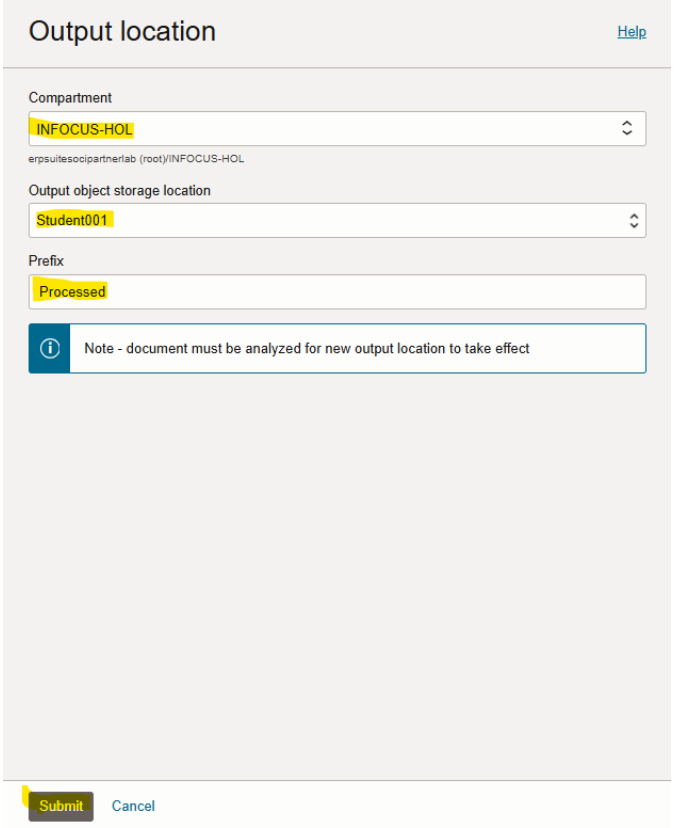
**1.** Click on the newly created model name **Studentxxx**



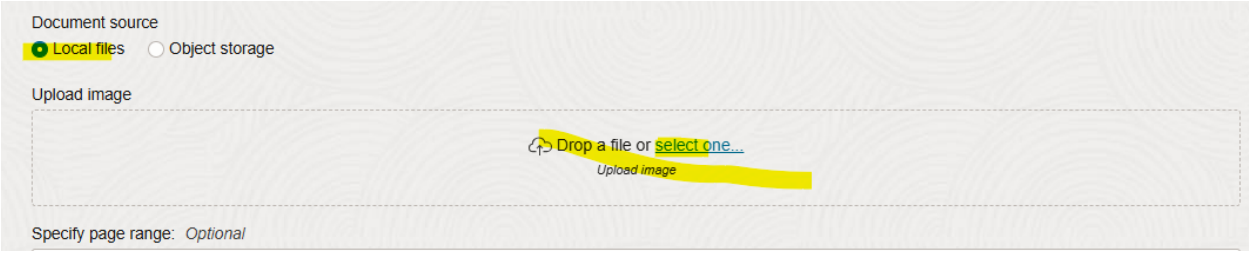
2. Click on **Select to change output location**



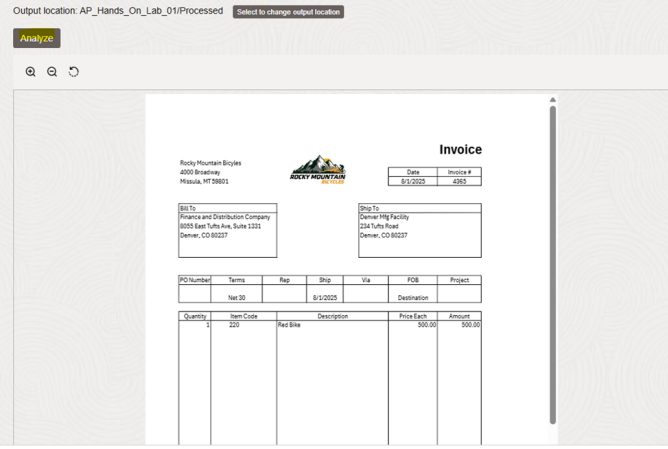

3. On the right-hand side under output location
   a. Select the **INFOCUS-HOL** from the drop-down list
   **b.** Select your bucket **Studentxxx** from the drop-down list
   c. Under Prefix type **Processed**
   d. Click **Submit**



4. Select **Local Files and upload** the files you trained from local storage




5. Once uploaded Select **Analyze**




On the right-hand side hit down arrow to see the json **Request** is what goes in the rest call of orchestration that kicks off document understanding


On the right-hand side hit down arrow to see the json **Response** is what we get back from the orchestration

Response

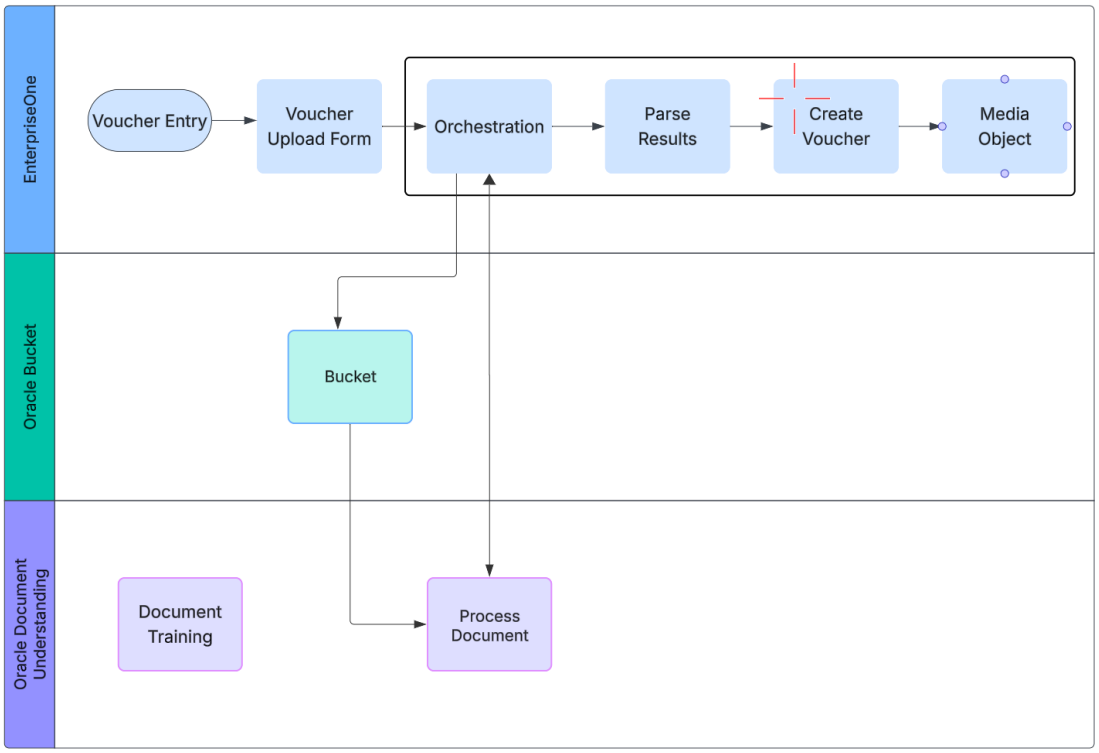
JSON Copy


```
{
  "documentMetadata": {
    "pageCount": 1,
    "mimeType": "application/pdf"
  },
  "pages": [
    {
      "pageNumber": 1,
      "dimensions": {
        "width": 2550,
        "height": 3301,
        "unit": "PIXEL"
      },
      "words": [
        {
          "text": "Invoice",
          "confidence": 0.99979395,
          "boundingPolygon": {
            "normalizedVertices": [
              {
                "x": 0.7301960784313726,
                "y": 0.09481975159042715
              },
              {
                "x": 0.8364705882352941,
                "y": 0.09481975159042715
```

## Understanding the Orchestrations

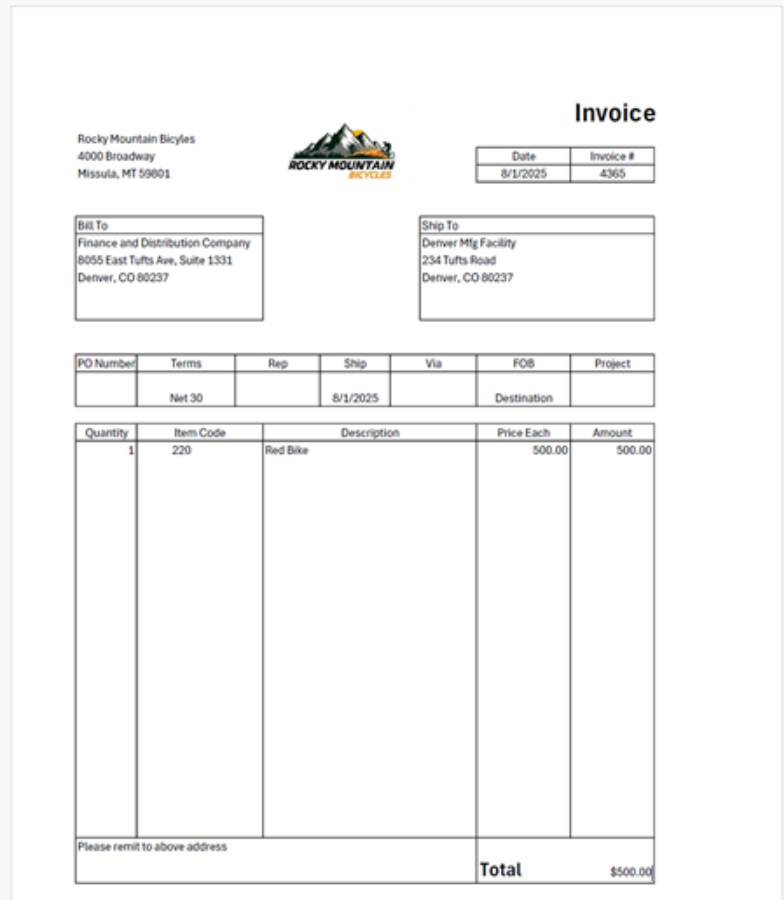
Overview

The Orchestration is being triggered from Standard Voucher Entry (P0411). The process flow is illustrated below. Once an invoice is submitted, the Orchestration sends the pdf file to an OCI Bucket. The file is then read by Document Understanding and parsed according to the training. The results of this process are returned to the Orchestration where validation occurs.  If the data returned is acceptable the Voucher Form Request creates the voucher.



Executing the Orchestration from Voucher Entry.

Example invoice document to load.



Invoice

Rocky Mountain Bicyles
4000 Broadway
Missula, MT 59801

| Date | Invoice # |
|---|---|
| 8/1/2025 | 4365 |

| Bill To |
|---|
| Finance and Distribution Company<br>8055 East Tufts Ave, Suite 1331<br>Denver, CO 80237 |

| Ship To |
|---|
| Denver Mfg Facility<br>234 Tufts Road<br>Denver, CO 80237 |

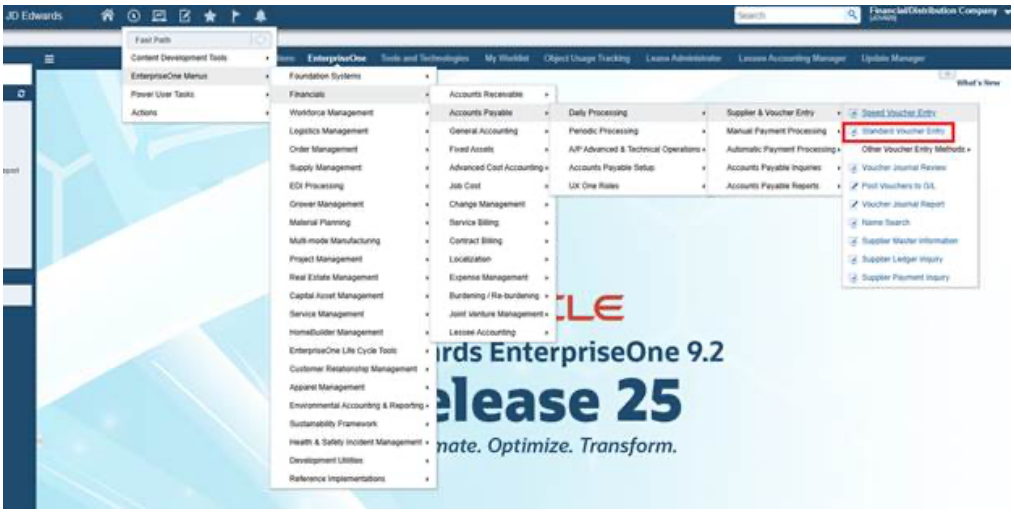
| PO Number | Terms | Rep | Ship | Via | FOB | Project |
|---|---|---|---|---|---|---|
| | Net 30 | | 8/1/2025 | | Destination | |

| Quantity | Item Code | Description | Price Each | Amount |
|---|---|---|---|---|
| 1 | 220 | Red Bike | 500.00 | 500.00 |

Please remit to above address

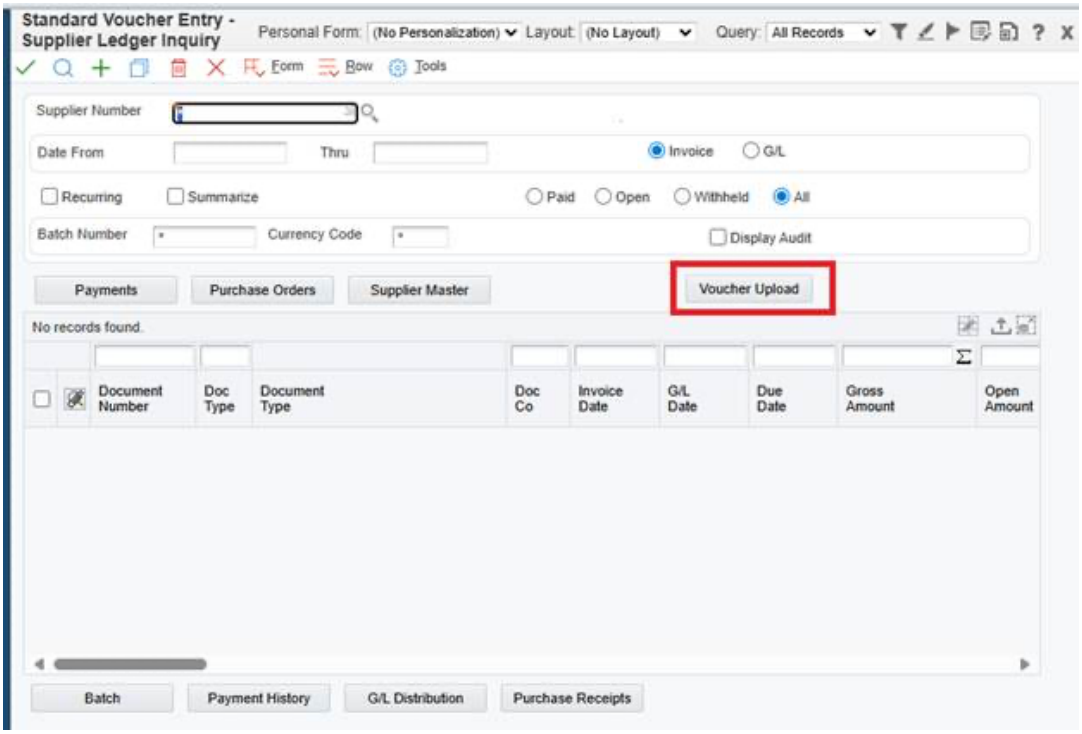
Total $500.00

1. Take the menu path to Standard Voucher Entry.

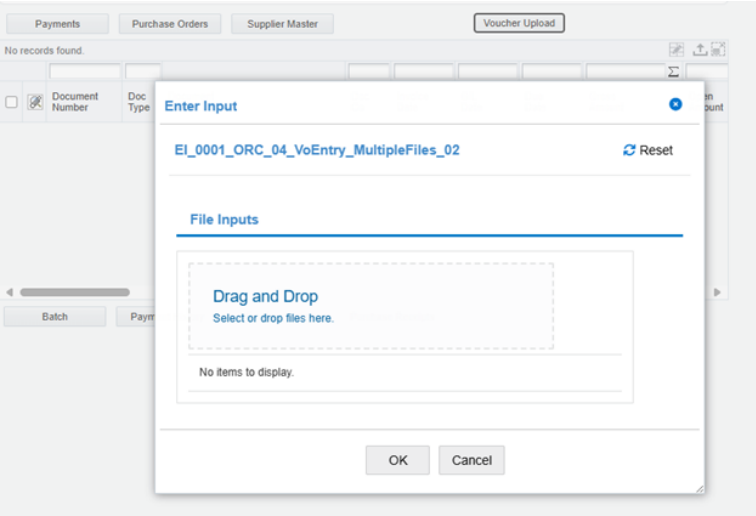



A User Defined Object has been defined and a Voucher Upload button has been created. This button executes the Orchestration.

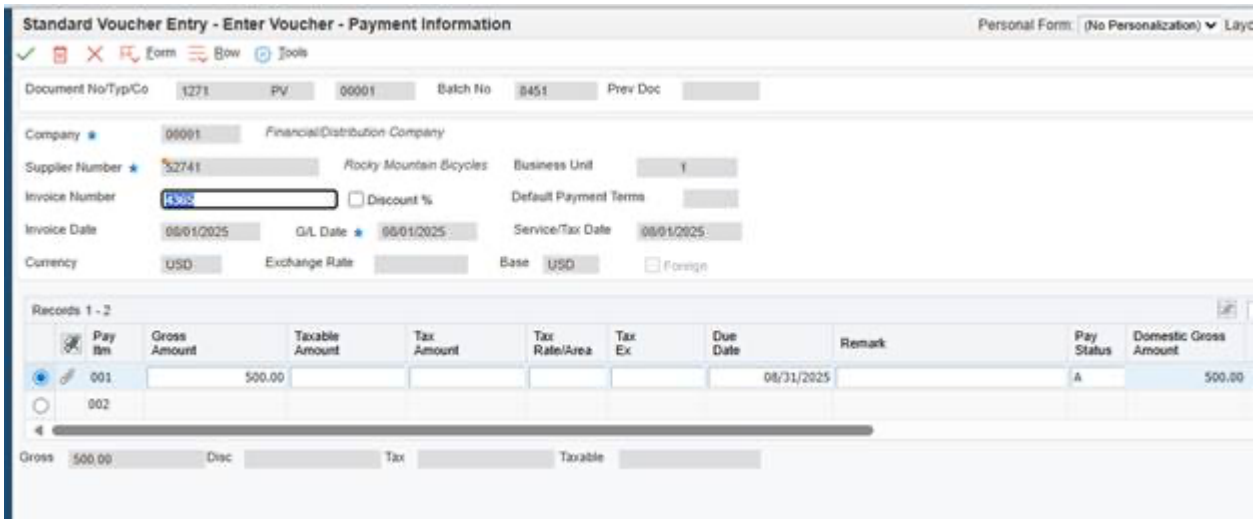
2. Click Voucher Upload



**3.** Drag and Drop, or select the invoice pdf file. Click 'OK'.

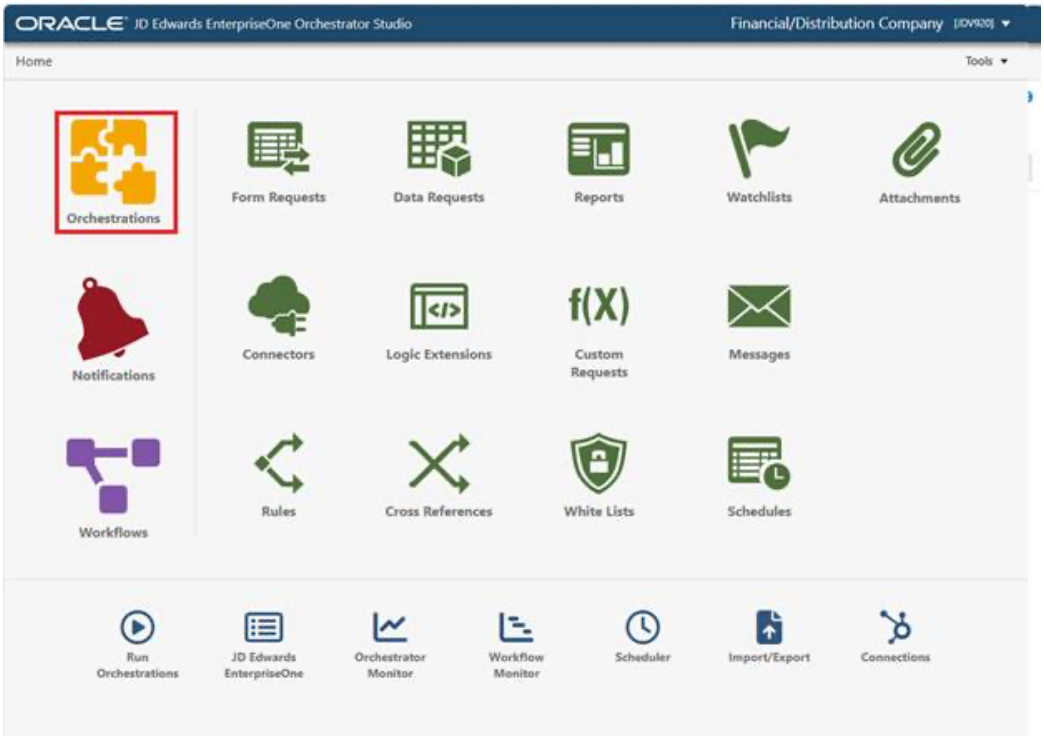



**4.** The Orchestration will execute, and the new voucher will display after pressing the 'find' button.

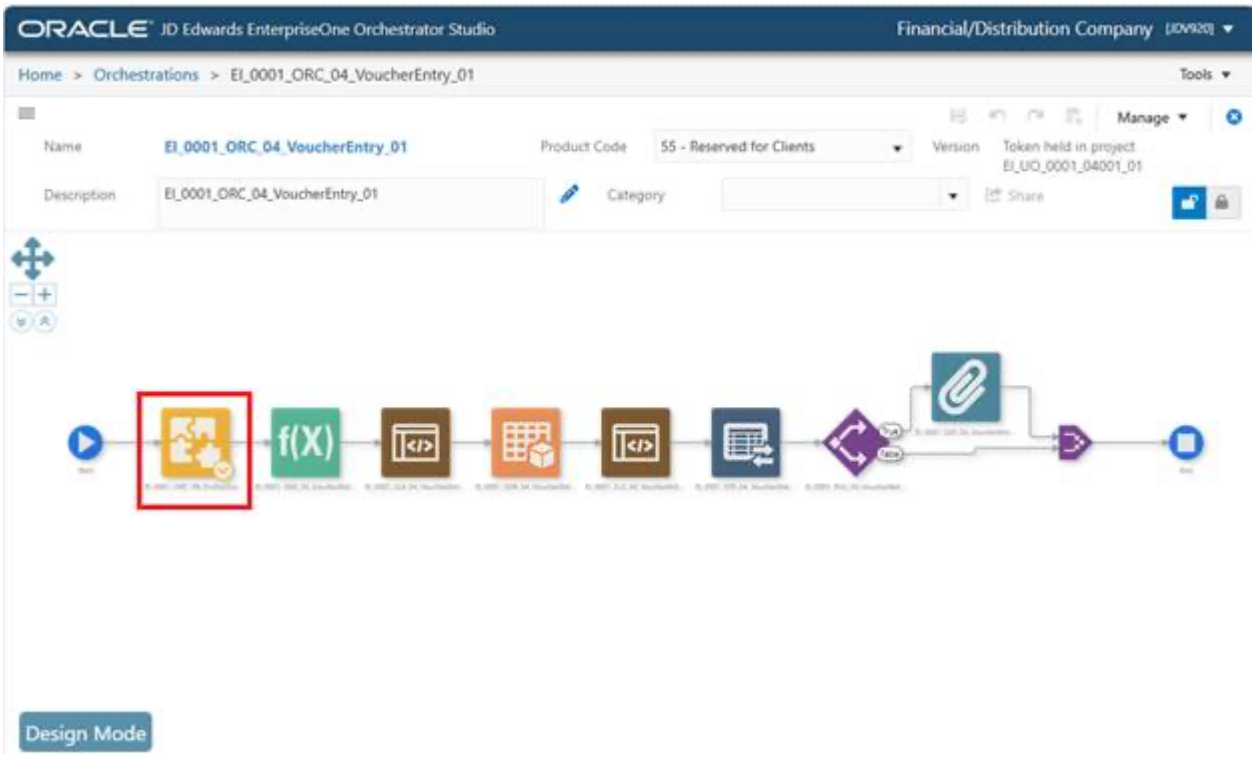


**Orchestration Architecture**

Below is an overview of an ERPSuites solution to process an invoice using Oracle Document Understanding and the creation of a new Voucher.



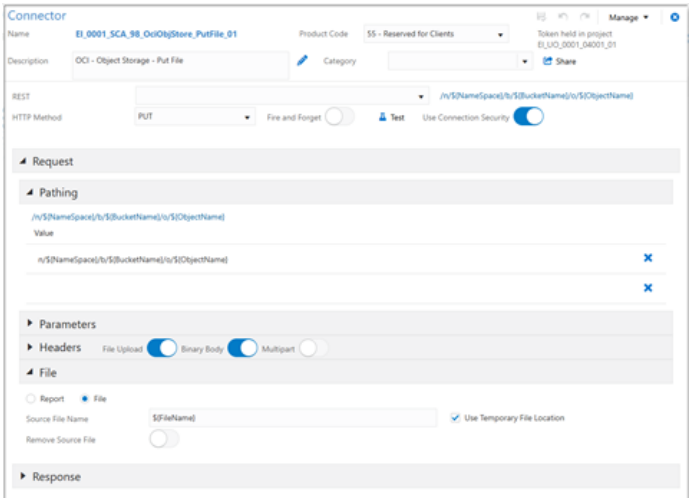
This Orchestration does the work of sending the invoice to Oracle Document Understanding, parsing the results, executing Voucher Entry, and loading the invoice document to a Media Object.



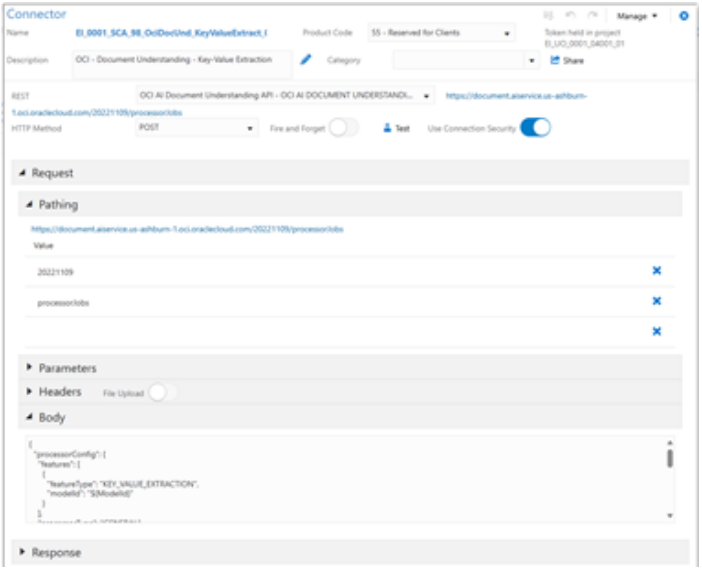

Step 1 – a Sub Orchestration to trigger the Oracle Document Understanding

The sub-orchestration performs the following 5 tasks:

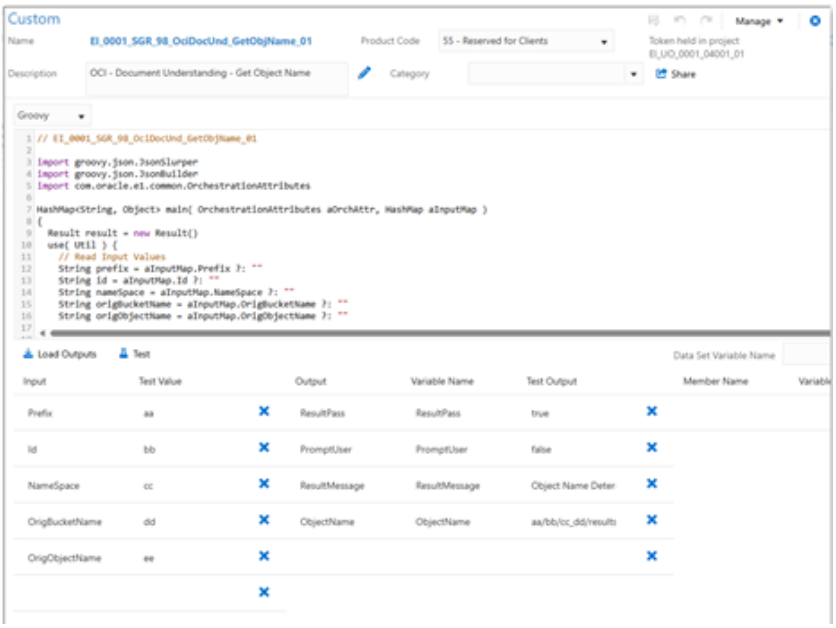
1.1 Send the invoice to an OCI Bucket



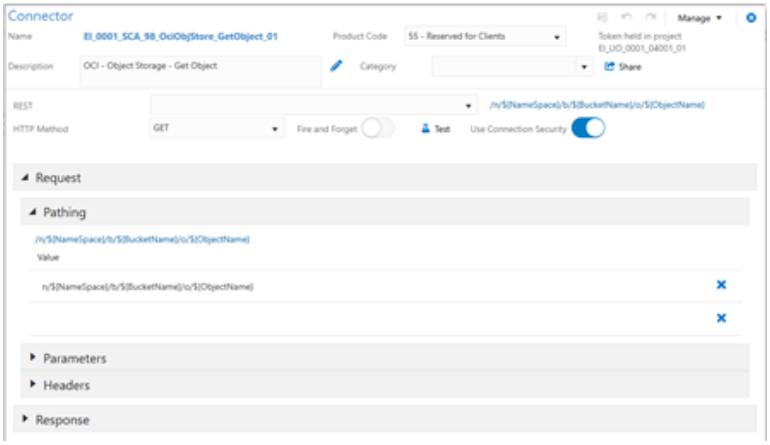

1.2 Trigger the Oracle Document Understanding service and supplying the job to run.

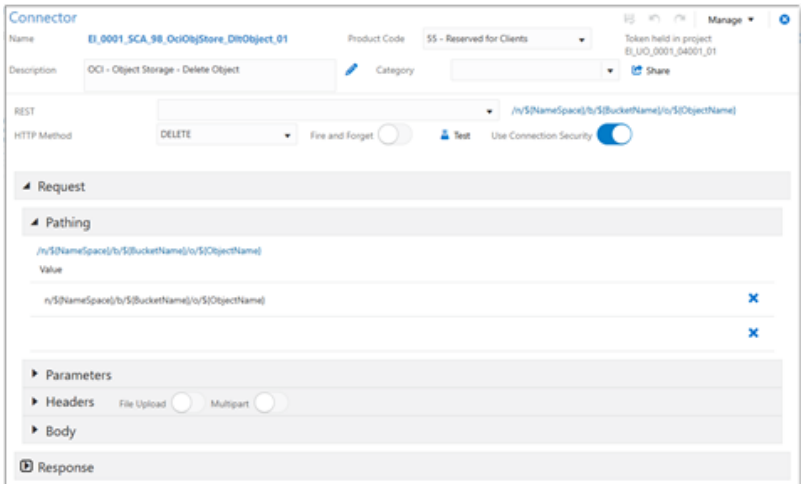
### 1.3 Building a string for retrieving the results


### 1.4 Getting the results from Oracle Document Understanding.

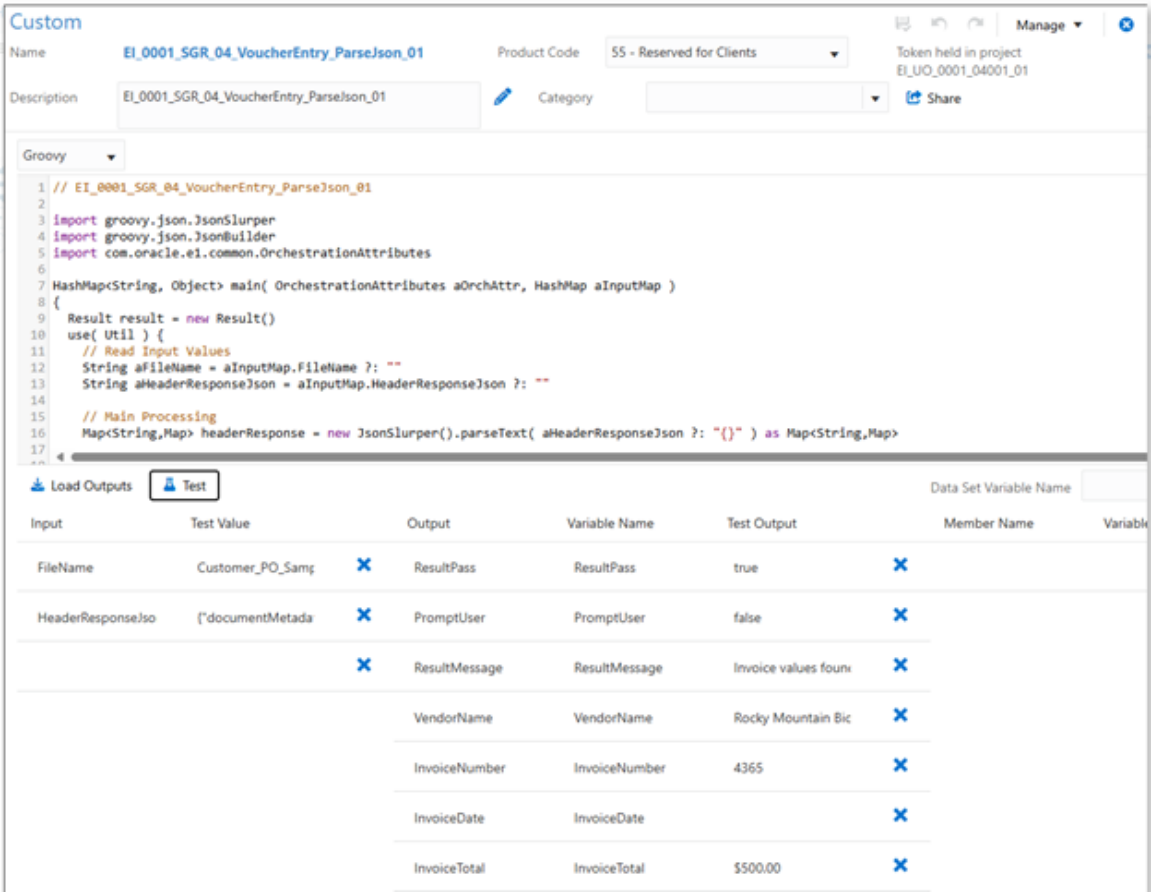



1.5 Deleting the results from the Oracle Bucket

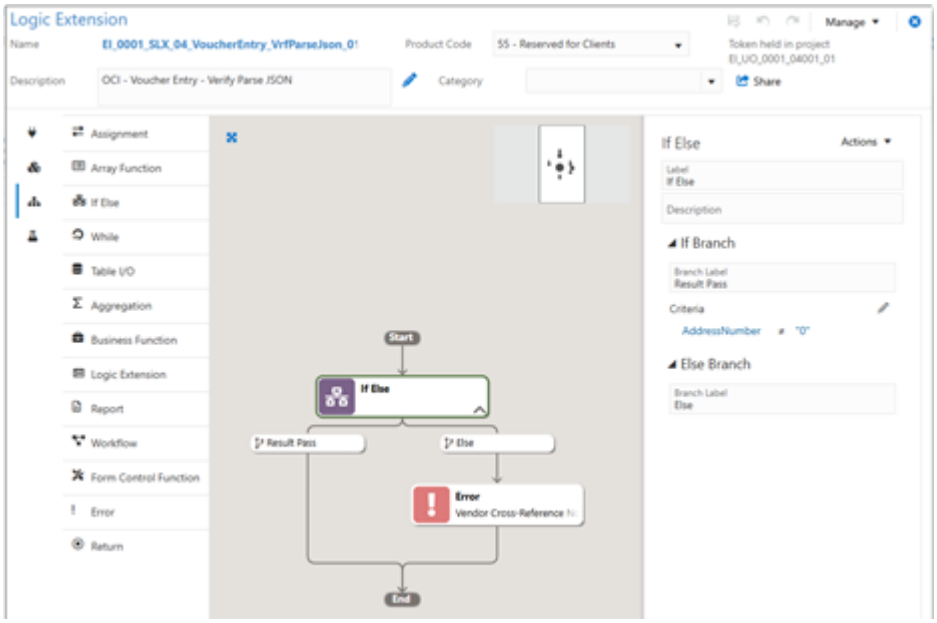



Step 2 – of the main Orchestration – Parsing the results of Document Understanding.

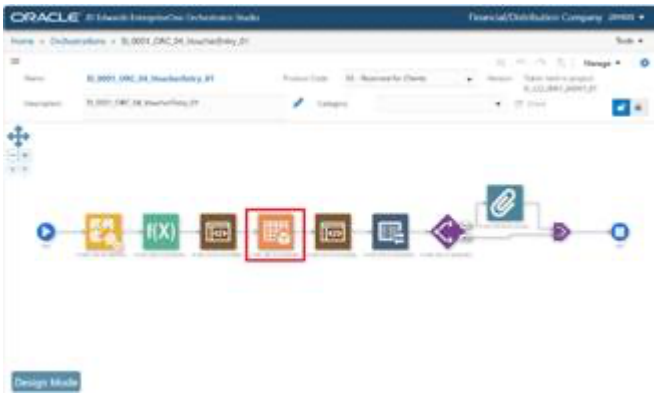
This Groovy Script reads the results passed back from Oracle and retrieves the relevant fields.



Step 3 – Verify an Invoice Number was returned.

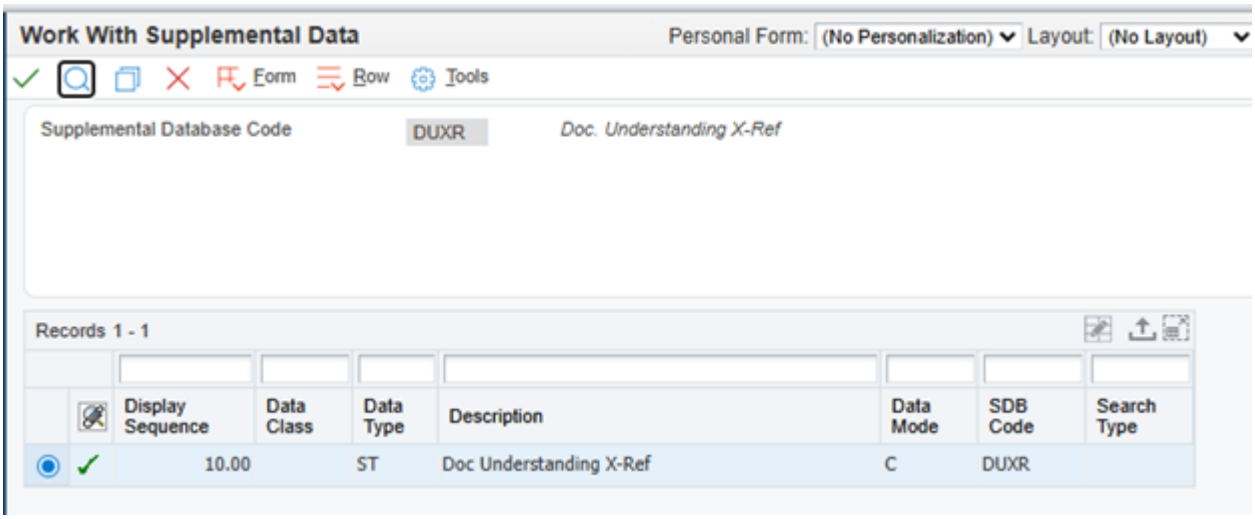


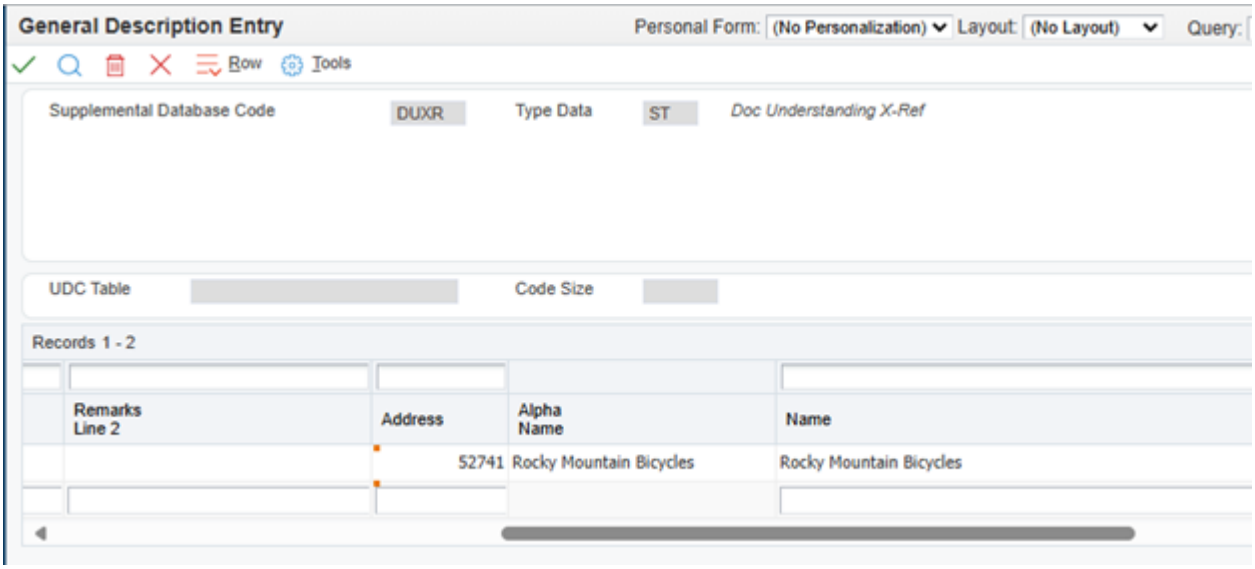
Step 4 – Search for the Vendor's Address Book Number



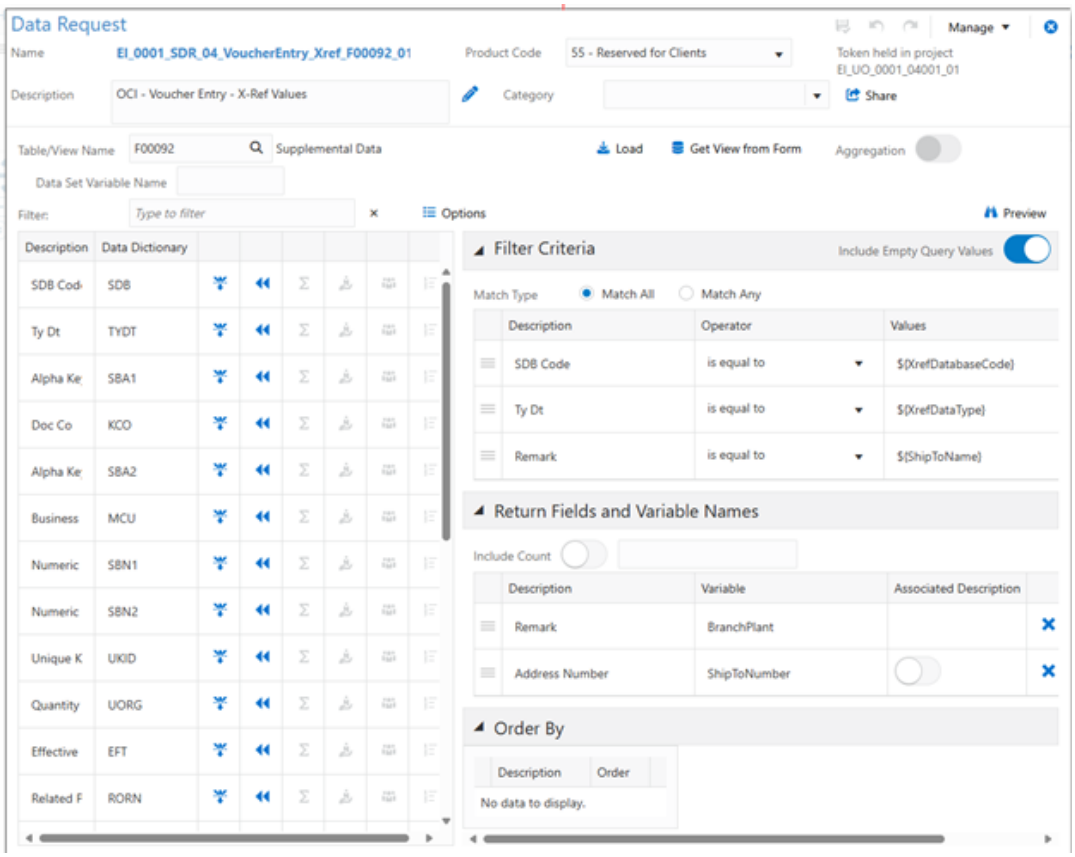
Background for this search being performed in Step 4.

The name of the vendor on the invoice might not match the value stored in the Address Book. A cross-reference table is required to find a related Address Book Number based on the name. We used the Supplemental Data applications (P00091/P00092) to set up Cross-Reference Information. In this case Rocky Mountain Bicycles returns Address Book Number 52741.

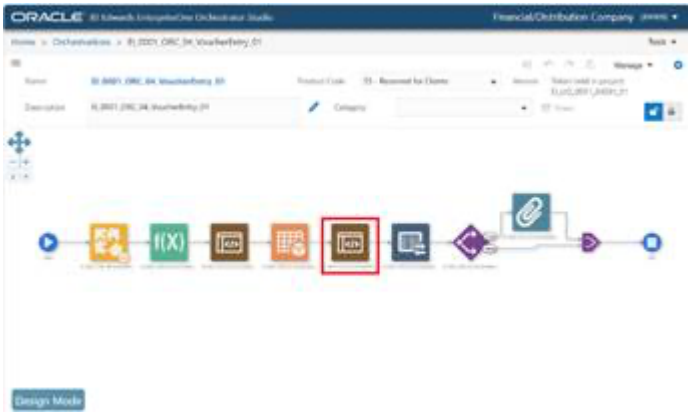




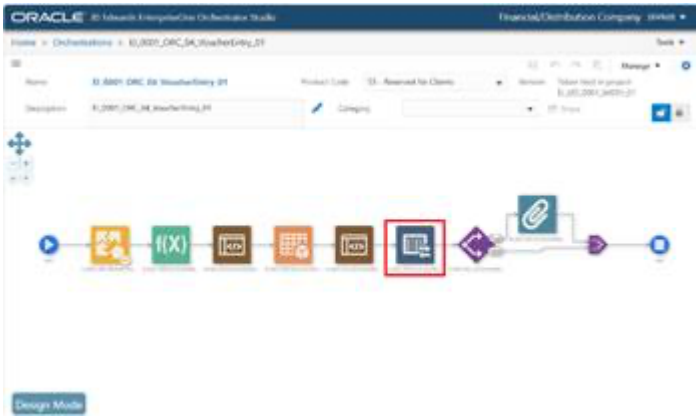

The Data Request to perform the search.

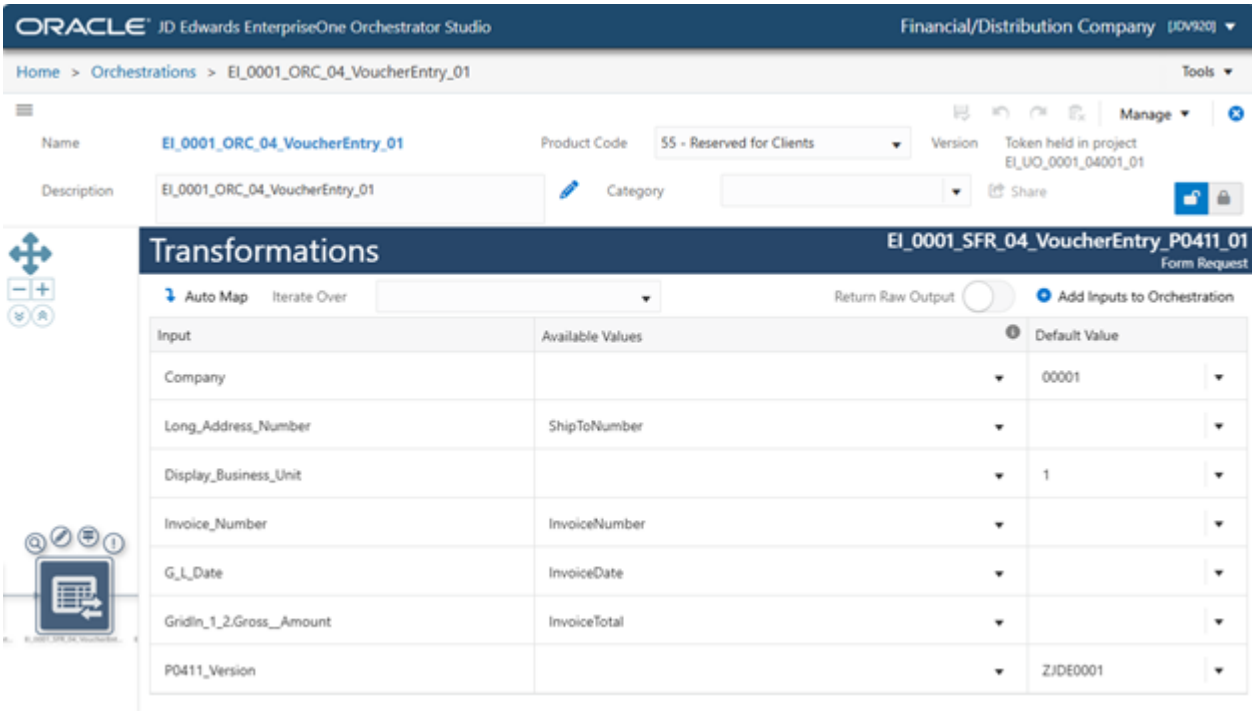


Step 5 – Verify an Address Book Number was retrieved. This uses the same logic extension as Step 3.

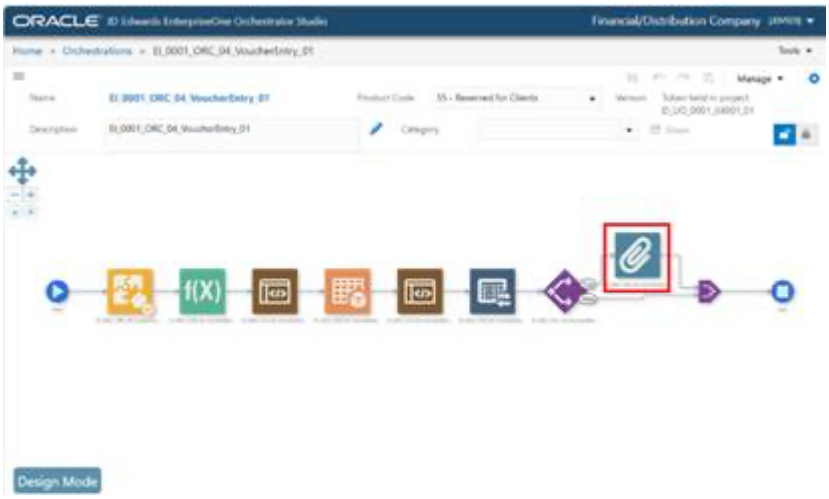


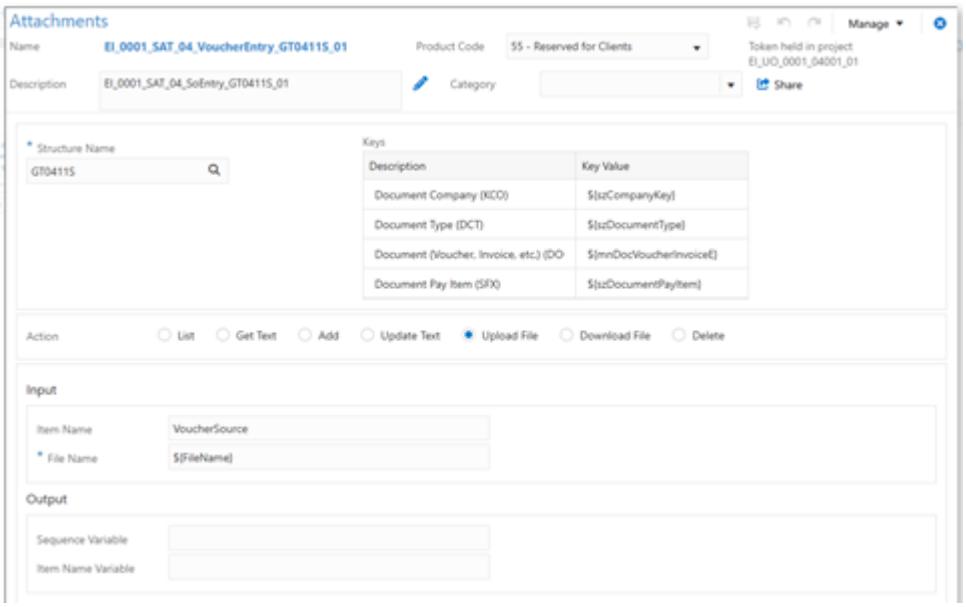
Step 6 – Call the Voucher Entry application Form Request.





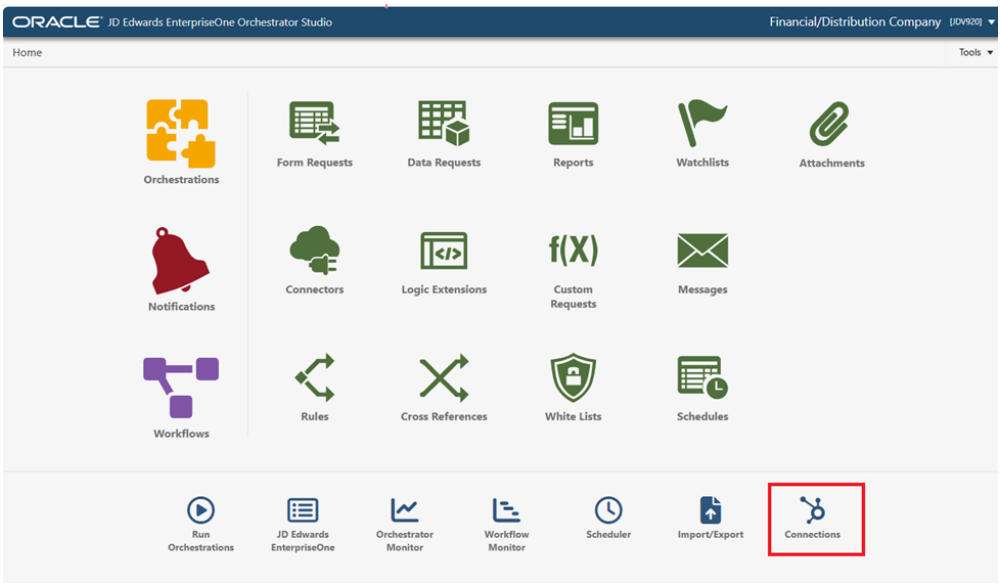

Step 7 – Save the Document into a Voucher Media Object.




Attachments
Manage
Name
EI_0001_SAT_04_VoucherEntry_GT0411S_01
Product Code
55 - Reserved for Clients
Token held in project
EI_UO_0001_04001_01
Description
EI_0001_SAT_04_SoEntry_GT0411S_01
Category
Share
* Structure Name
GT0411S
Keys
Description
Key Value
Document Company (KCO)
${szCompanyKey}
Document Type (DCT)
${szDocumentType}
Document (Voucher, Invoice, etc.) (DO
${mnDocVoucherInvoiceE}
Document Pay Item (SFX)
${szDocumentPayItem}
Action
List
Get Text
Add
Update Text
Upload File
Download File
Delete
Input
Item Name
VoucherSource
* File Name
${FileName}
Output
Sequence Variable
Item Name Variable

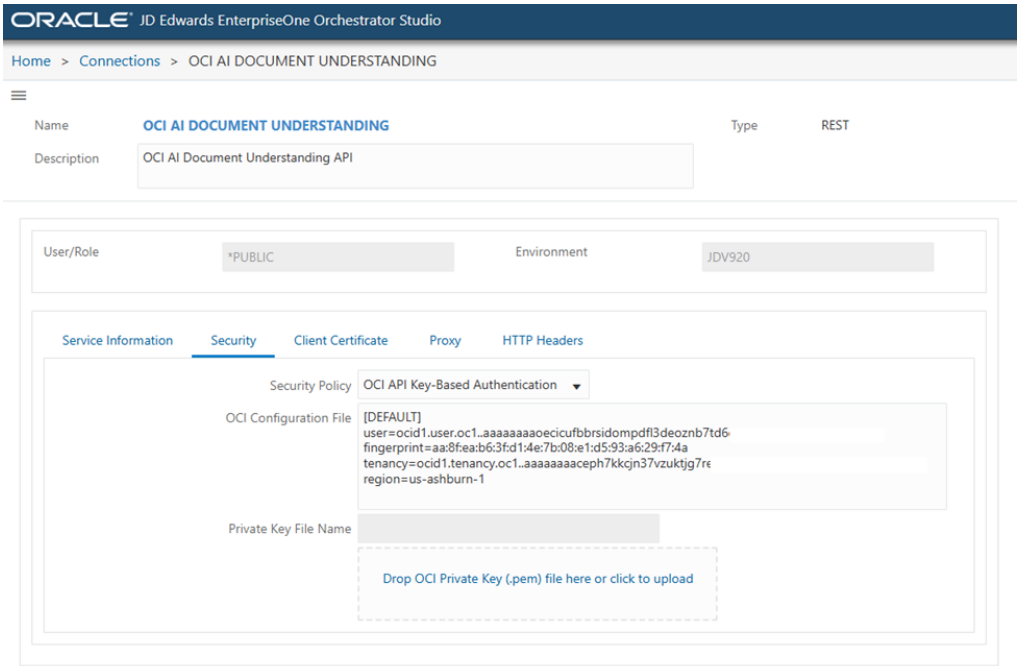
## OCI Connection Details

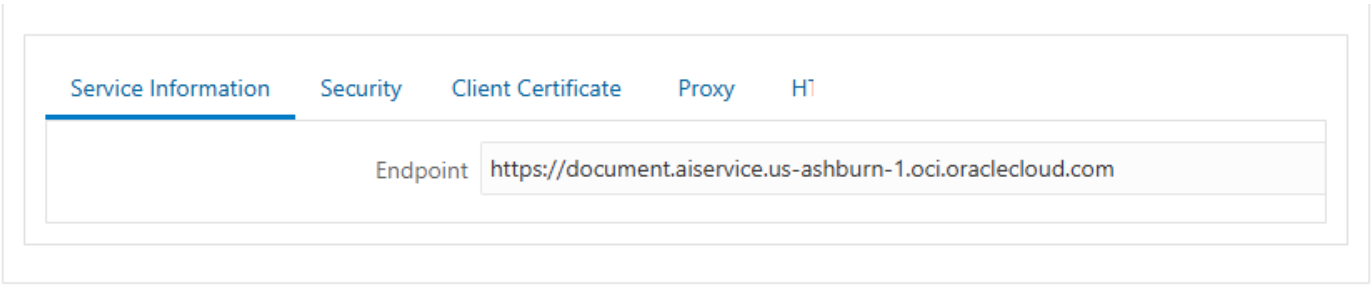
**Note: JDE version required for this example is:** version 9.2.8.2




Connection to Document Understanding – Sending files into Document Understanding

- The configuration file and PEM file will be unique to each tenancy






Connection to Object Storage – Retrieving Bucket data

- The configuration file and PEM file will be unique to each tenancy