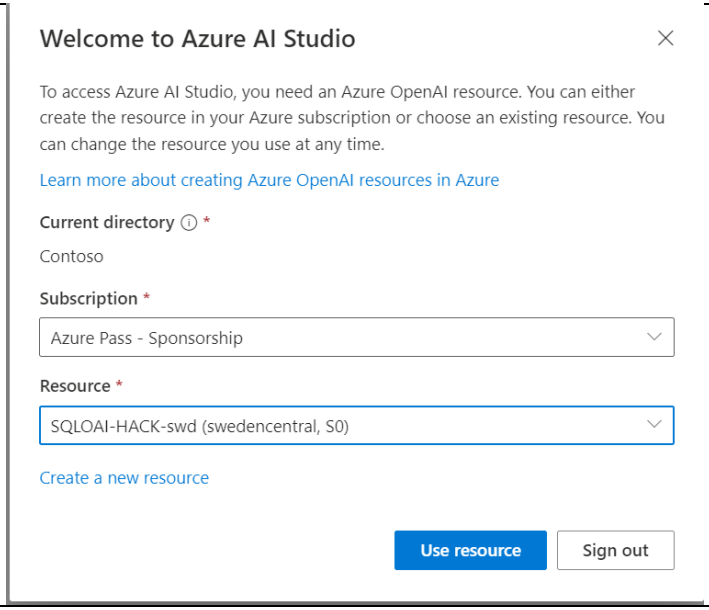
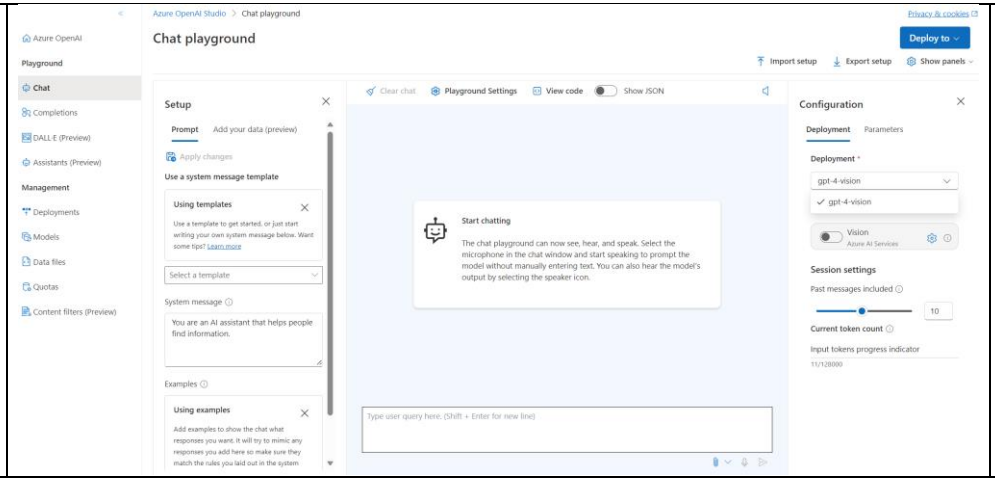
Lab 5 - Image Analysis with GPT 5 Turbo Vision

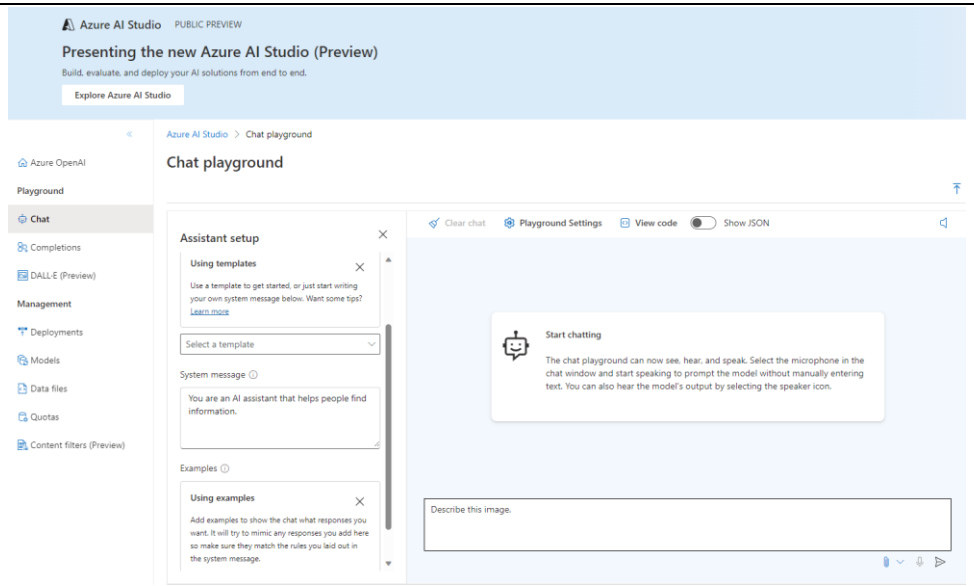
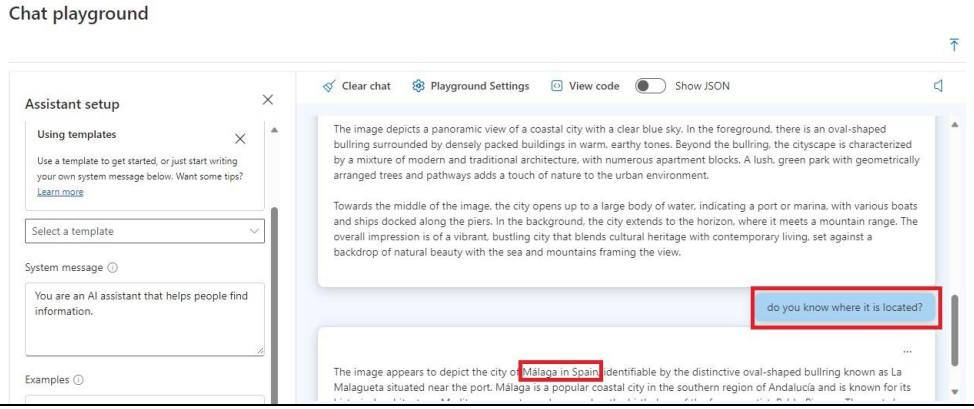
Contents

1. GPT-4 with Vision in Azure OpenAI Studio

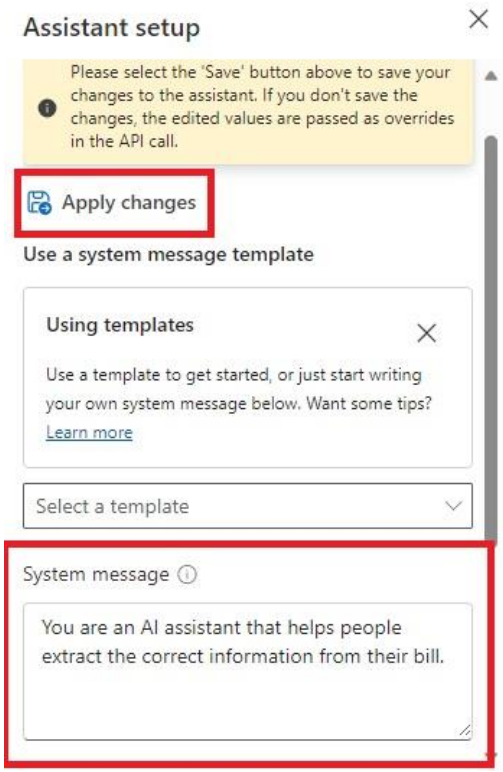
Start exploring GPT-4 Turbo with Vision capabilities with a no-code approach through Azure OpenAI Studio.

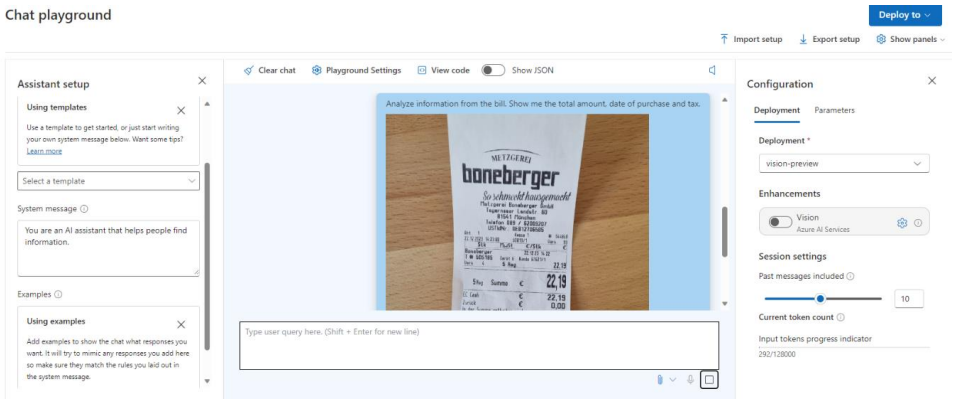
Narrative	Screenshot	Notes
Navigate to the Azure OpenAI studio again: **https://oai.azure.com/portal** Select the following: • Directory: ***Contoso*** • Subscription: ***Azure Pass – Sponsorship*** As the **Azure OpenAI resource**, choose the one according to your TEAM number: • TEAM 01 – 07: SQLOAI-HACK-swd • TEAM 08 – 14: SQLOAI-HACK-wus • TEAM 15 – 20: SQLOAI-HACK-jpe		

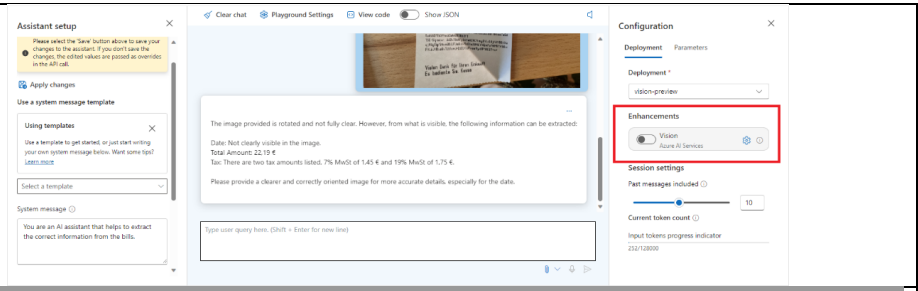
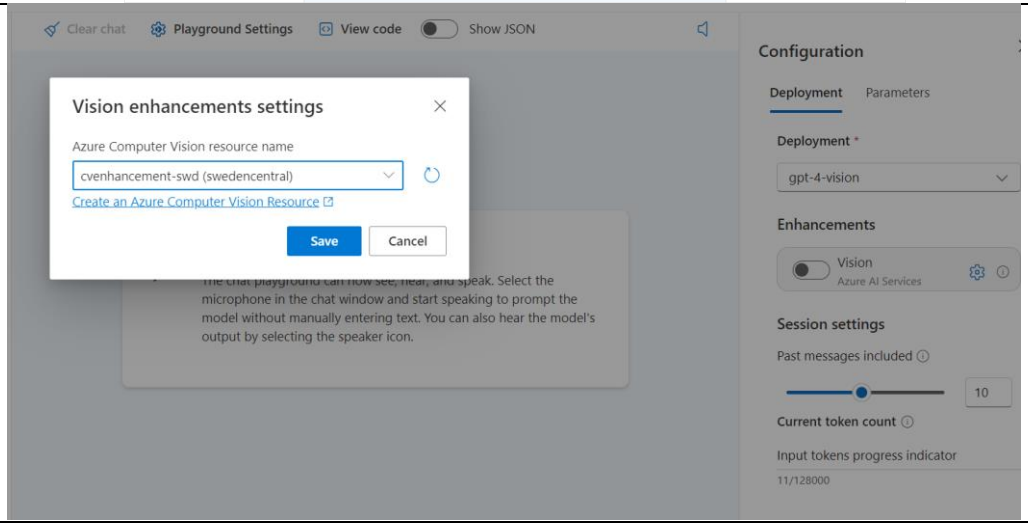
Navigate to **Chat playground**. Switch your deployment model to **gpt-4-vision** on the right side under “Deployments”.		

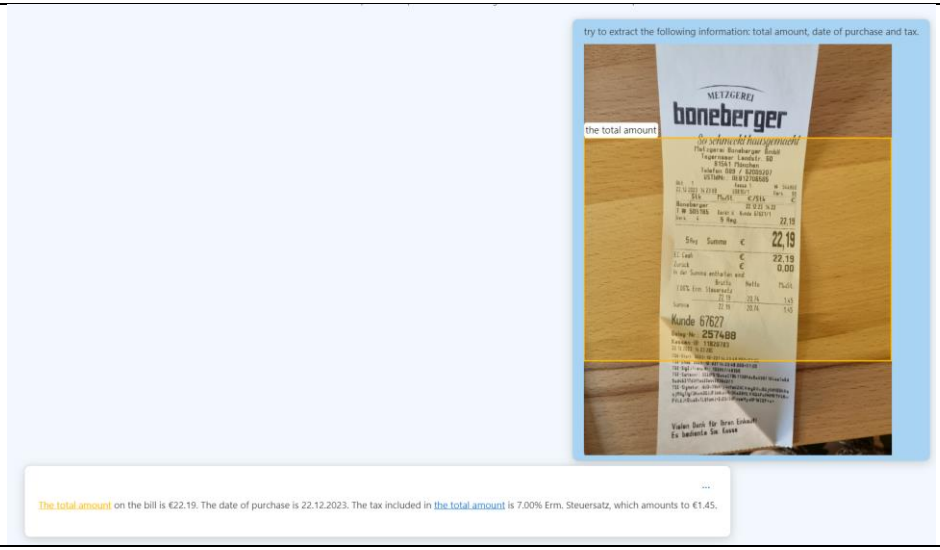
In the **Assistant setup** pane, provide this System Message to guide the assistant: *"You are an AI assistant that helps people find information."* You should tailor the System Message to the image or scenario that you're uploading. **Save your changes**, and when prompted to confirm updating the system message, select **Continue**. In the **Chat session** pane, enter a text prompt like "*Describe this image,"* and **upload an image** with the attachment button from the lab 5 images folder. You can use a different text prompt for your use case. Then select **Send**.	
Observe the output provided. Consider asking follow-up questions related to the analysis of your image to learn more. For example, if it is a picture of city, you can ask to provide you with the name of the city and country where it is located. Choose any image you would like to analyse and play with the prompt's formulation.	

In the next part of the 1. Exercise section, we would like to work with the data that it might be more difficult to analyze.

Narrative	Screenshot	Notes
For the exercise, stay in your Azure AI Studio and select vision-preview as a model in your Chat playground. We will work with information extraction from the bills. You can adjust system message to guide your model what exactly is needed to be performed. For example, in the **system message** you can specify the following guidance: *You are an AI assistant that helps to extract the correct information from the bills.* Click **Apply Changes** in the Assistant Setup section. Now upload an image. Navigate to your folder Hack Repository, Select Travel-Expense-Assistant, go to the images folder and select folder bills. Select any of the bills and try to extract the following information: total amount, date of purchase and tax.		

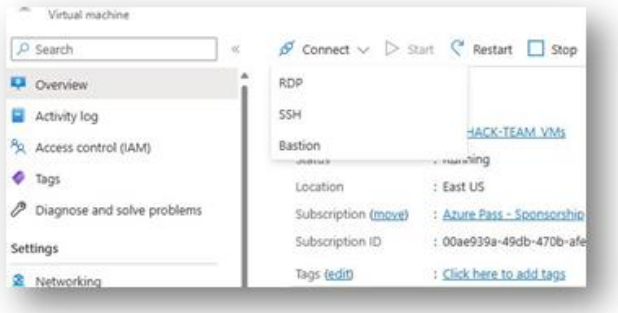
When you search an image to upload, then navigate to your folder Hack Repository, Select Travel-Expense-Assistant, go to the images folder and select folder bills. Select any of the bills and try to extract the following information: total amount, date of purchase and tax. Upload an image.		
You can get the following output as an answer. *The image provided is rotated and not fully clear. However, from what is visible, the following information can be extracted:* *Date: Not clearly visible in the image.* *Total Amount: 22,19 €* *Tax: There are two tax amounts listed, 7% MwSt of 1,45 € and 19% MwSt of 1,75 €.* *Please provide a clearer and correctly oriented image for more accurate details, especially for the date.*	The image provided is rotated and not fully clear. However, from what is visible, the following information can be extracted: Date: Not clearly visible in the image. Total Amount: 22,19 € Tax: There are two tax amounts listed, 7% MwSt of 1,45 € and 19% MwSt of 1,75 €. Please provide a clearer and correctly oriented image for more accurate details, especially for the date.	

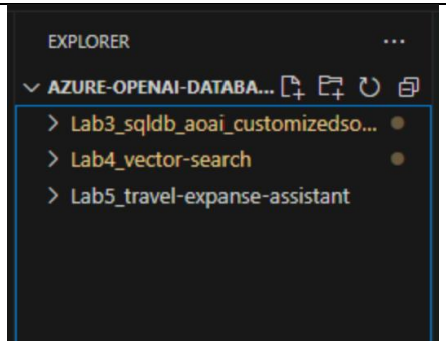
Now let’s check the results when we enable **Vision enhancement**.		
Select an Azure Computer vision resource. The screenshot shows an example.		

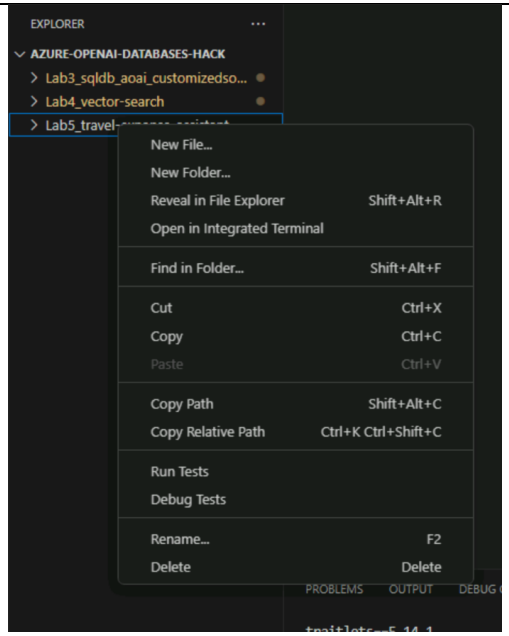
Upload the same image with the same prompt and compare the current and previous results. You can also click on the **highlighted parts** in the output which will show the respective areas in the image. E.g. click on “total amount”.		

2. GPT-4 with Vision in the customized solution

Log into your Team VM

Narrative	Screenshot	Notes
If you are logged out of your Team VM, go to the **Azure Portal** and navigate to your dedicated Team VM. Click on ***Connect > Bastion***. Then use the following credentials: ***Username:*** *DemoUser* ***Password:*** *Demo@pass1234567*		

If you closed it, reopen **Visual Studio Code** with your already imported lab folders.				

Open the terminal for your Lab5_travel-expense-assistant by right clicking on the folder and selecting **Open in integrated terminal**. This will open a terminal navigated into the lab 5 folder for you.		
Check if the needed requirements are already installed with the command: ***pip install -r requirements.txt***		

In the lab 5 folder, open the file **Expense Report Extraction.py**

Now set the **credential values** within the application code.

Navigate to the following lines within the code that should be populated with the values of services we use for the Hack.

Insert **AOAI_API_BASE** and **AOAI_API_KEY** based on the information provided below.

Change the **SQL_DATABASE** to your team's value, e.g. TEAM01_LocalMasterDataDB.

Update **Azure Computer Vision endpoint and key** based on the information provided below.

Save the updates. The other values are already set up for you.

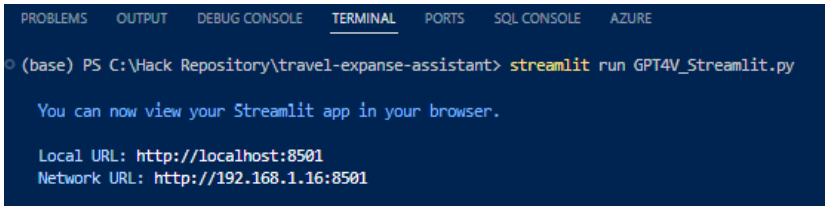
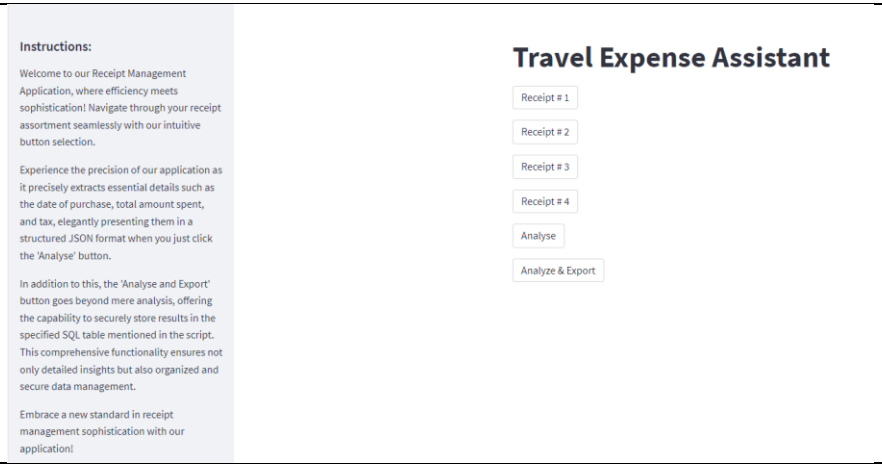
```
# Extracting environment variables. Adjust values according to the lab documentation.
AOAI_API_BASE = "..."
AOAI_API_KEY = "..."
AOAI_API_VERSION = "2023-12-01-preview"
AOAI_DEPLOYMENT = "gpt-4-vision"

# Database connection details. Adjust values according to the lab documentation.
DB_SERVER = "sqlhackmi-j754o5hum2r36.7a59bf01d694.database.windows.net"
DB_NAME = "TEAM02_LocalMasterDataDB"
DB_USERNAME = "DemoUser"
DB_PASSWORD = "Demo@pass1234567"
```

```
    "dataSources": [
        {
            "type": "AzureComputerVision",
            "parameters": {
                "endpoint": "...",
                "key": "...."
            }
        }
    ],
```

Azure OpenAI

Teams	AOAI API Base	AOAI API Key
TEAM 01 – 07	https://sqloai-hack-swd.openai.azure.com/	0c59924d91d24708bdbfd6d1f9333599
TEAM 08 – 14	https://sqloai-hack-wus.openai.azure.com/	44258e8588ea459e9466813bd1fb2b56
TEAM 15 – 20	https://sqloai-hack-jpe.openai.azure.com/	c877d65db48a4cb4905f319d7859e078

Azure Computer Vision	**Computer Vision Endpoint:** https://eastus.api.cognitive.microsoft.com/ **API key:** bbc010d375d24ee584b4af4993ded86b	
In **Visual Studio Code**, open a terminal and navigate to lab 5. To run the application from the command line: ***streamlit run '.\Expense Report Extraction.py'***		
A new browser window will be opened that shows our streamlit application.		

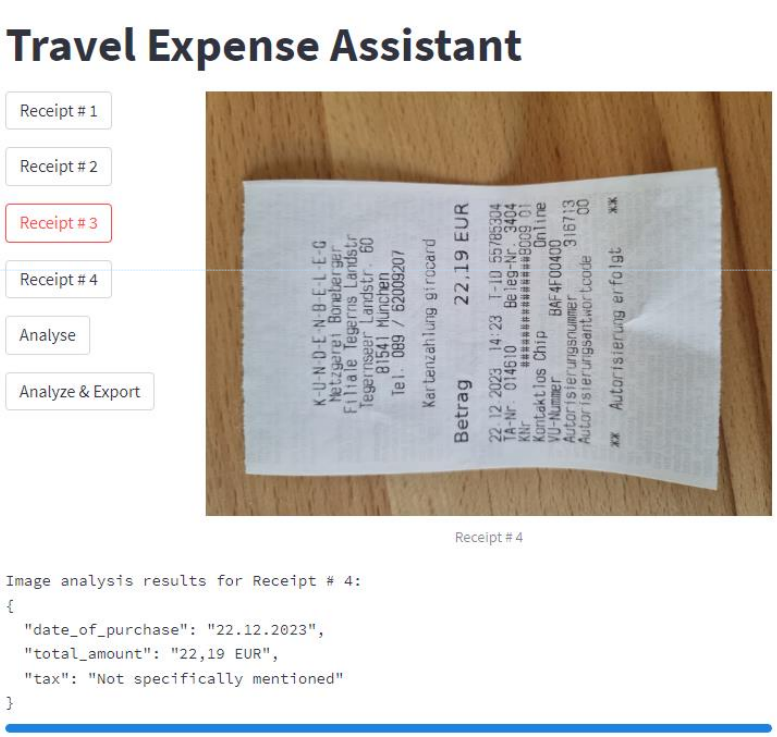
As it is mentioned in the description, you have the following options:

- Click through and check the bills that are stored locally that contain the bills to be analysed. You can find them in your folder: ***C:_OpenAI-SQL_\Azure-OpenAI-Databases-Hack\Lab5_travel-expanse-assistant\images \bills***

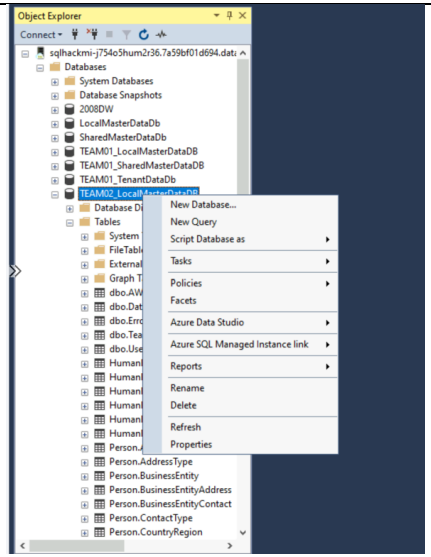
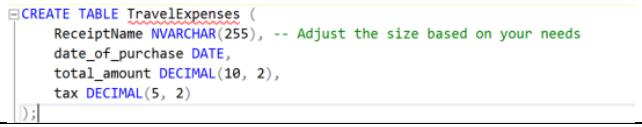
- With **Analyse** button you can explore the content of your bill with the help GPT-4-Turbo with Vision in your Azure OpenAI deployment. The following information will be presented to the user: total amount of the purchase, date of the purchase, and tax amount paid.

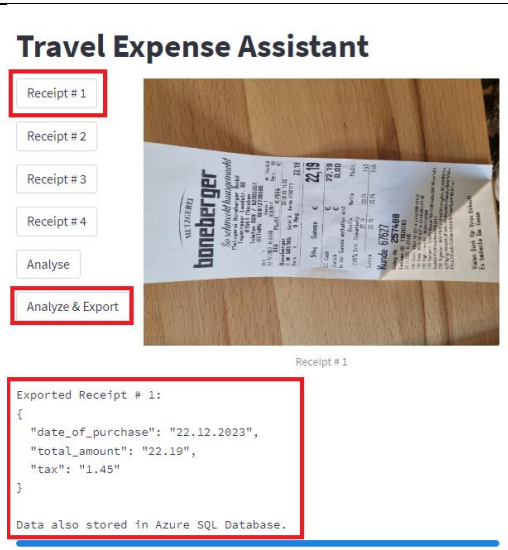
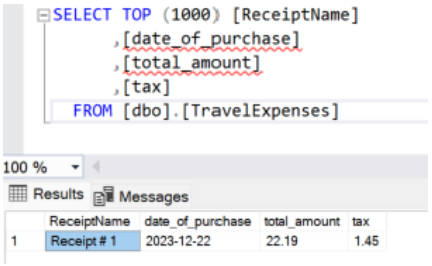
- **Analyse&Export** button allows the user to get the main information from the bill and to store the data in SQL Table.

To have our data exported to SQL Table, please perform the next step.



Commented [ML1]: update

In SSMS, **create a table** to store your analysed data in your Team DB. Open SSMS again and connect to the Azure SQL MI as before. Navigate to your TeamXX_LocalMasterDataDB Right click to run the following query: *CREATE TABLE TravelExpenses (ReceiptName NVARCHAR(255), date_of_purchase DATE, total_amount DECIMAL(10, 2), tax DECIMAL(5, 2));*	 	
Refresh and check if the table is already there.		

In the app, **select a Receipt** to be analyzed and export it to the newly created table.		
In SSMS, check if the data was inserted correctly to your Azure SQL DB. *SELECT TOP (1000) [ReceiptName]* *,[date_of_purchase]* *,[total_amount]* *,[tax]* *FROM [dbo].[TravelExpenses]*		
If it is needed, repeat the last two steps again to check the insert functionality.		