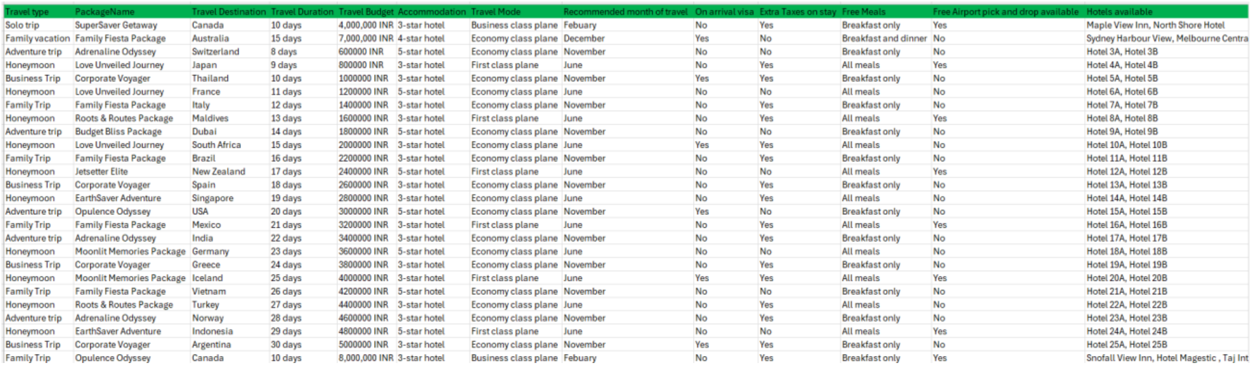
Project Name: Travel Planner

Motive of this project is to provide the best package from the database based on the user preferences.Database has got plethora of data which includes the data for different countries having information of different hotels , Travel type,PackageName,Travel Destination, Travel Duration,Travel Budget,Accommodation,Travel Mode,Recommended month of travel,On arrival visa,Extra Taxes on stay,Free Meals,Free Airport pick and drop available,Hotels available,Nearby attractions,Pets allowed in hotel,Swimming pools in hotel,Free amenities in hotel,Summary.

Screenshot from the database



Travel type	PackageName	Travel Destination	Travel Duration	Travel Budget	Accommodation	Travel Mode	Recommended month of travel	On arrival visa	Extra Taxes on stay	Free Meals	Free Airport pick and drop available	Hotels available
Solo trip	SuperSaver Getaway	Canada	10 days	4,000,000 INR	3-star hotel	Business class plane	Febuary	No	Yes	Breakfast only	Yes	Maple View Inn, North Shore Hotel
Family vacation	Family Fiesta Package	Australia	15 days	7,000,000 INR	4-star hotel	Economy class plane	December	Yes	No	Breakfast and dinner	No	Sydney Harbour View, Melbourne Centra
Adventure trip	Adrenaline Odyssey	Switzerland	8 days	600000 INR	5-star hotel	Economy class plane	November	No	No	Breakfast only	No	Hotel 3A, Hotel 3B
Honeymoon	Love Unveiled Journey	Japan	9 days	800000 INR	3-star hotel	First class plane	June	No	Yes	All meals	Yes	Hotel 4A, Hotel 4B
Business Trip	Corporate Voyager	Thailand	10 days	1000000 INR	3-star hotel	Economy class plane	November	Yes	Yes	Breakfast only	No	Hotel 5A, Hotel 5B
Honeymoon	Love Unveiled Journey	France	11 days	1200000 INR	5-star hotel	Economy class plane	June	No	No	All meals	No	Hotel 6A, Hotel 6B
Family Trip	Family Fiesta Package	Italy	12 days	1400000 INR	3-star hotel	Economy class plane	November	No	Yes	Breakfast only	No	Hotel 7A, Hotel 7B
Honeymoon	Roots & Routes Package	Maldives	13 days	1600000 INR	3-star hotel	First class plane	June	No	Yes	All meals	Yes	Hotel 8A, Hotel 8B
Adventure trip	Budget Bliss Package	Dubai	14 days	1800000 INR	5-star hotel	Economy class plane	November	No	No	Breakfast only	No	Hotel 9A, Hotel 9B
Honeymoon	Love Unveiled Journey	South Africa	15 days	2000000 INR	3-star hotel	Economy class plane	June	Yes	Yes	All meals	No	Hotel 10A, Hotel 10B
Family Trip	Family Fiesta Package	Brazil	16 days	2200000 INR	3-star hotel	Economy class plane	November	No	Yes	Breakfast only	No	Hotel 11A, Hotel 11B
Honeymoon	Jetsetter Elite	New Zealand	17 days	2400000 INR	5-star hotel	First class plane	June	No	No	All meals	Yes	Hotel 12A, Hotel 12B
Business Trip	Corporate Voyager	Spain	18 days	2600000 INR	3-star hotel	Economy class plane	November	No	Yes	Breakfast only	No	Hotel 13A, Hotel 13B
Honeymoon	EarthSaver Adventure	Singapore	19 days	2800000 INR	3-star hotel	Economy class plane	June	No	Yes	All meals	No	Hotel 14A, Hotel 14B
Adventure trip	Opulence Odyssey	USA	20 days	3000000 INR	5-star hotel	Economy class plane	November	Yes	No	Breakfast only	No	Hotel 15A, Hotel 15B
Family Trip	Family Fiesta Package	Mexico	21 days	3200000 INR	3-star hotel	First class plane	June	No	Yes	All meals	Yes	Hotel 16A, Hotel 16B
Adventure trip	Adrenaline Odyssey	India	22 days	3400000 INR	3-star hotel	Economy class plane	November	No	Yes	Breakfast only	No	Hotel 17A, Hotel 17B
Honeymoon	Moonlit Memories Package	Germany	23 days	3600000 INR	5-star hotel	Economy class plane	June	No	No	All meals	No	Hotel 18A, Hotel 18B
Business Trip	Corporate Voyager	Greece	24 days	3800000 INR	3-star hotel	Economy class plane	November	No	Yes	Breakfast only	No	Hotel 19A, Hotel 19B
Honeymoon	Moonlit Memories Package	Iceland	25 days	4000000 INR	3-star hotel	First class plane	June	Yes	Yes	All meals	Yes	Hotel 20A, Hotel 20B
Family Trip	Family Fiesta Package	Vietnam	26 days	4200000 INR	5-star hotel	Economy class plane	November	No	No	Breakfast only	No	Hotel 21A, Hotel 21B
Honeymoon	Roots & Routes Package	Turkey	27 days	4400000 INR	3-star hotel	Economy class plane	June	No	Yes	All meals	No	Hotel 22A, Hotel 22B
Adventure trip	Adrenaline Odyssey	Norway	28 days	4600000 INR	3-star hotel	Economy class plane	November	No	Yes	Breakfast only	No	Hotel 23A, Hotel 23B
Honeymoon	EarthSaver Adventure	Indonesia	29 days	4800000 INR	5-star hotel	First class plane	June	No	No	All meals	Yes	Hotel 24A, Hotel 24B
Business Trip	Corporate Voyager	Argentina	30 days	5000000 INR	3-star hotel	Economy class plane	November	Yes	Yes	Breakfast only	No	Hotel 25A, Hotel 25B
Family Trip	Opulence Odyssey	Canada	10 days	8,000,000 INR	3-star hotel	Business class plane	Febuary	No	Yes	Breakfast only	Yes	Snofall View Inn, Hotel Magestic , Taj Int

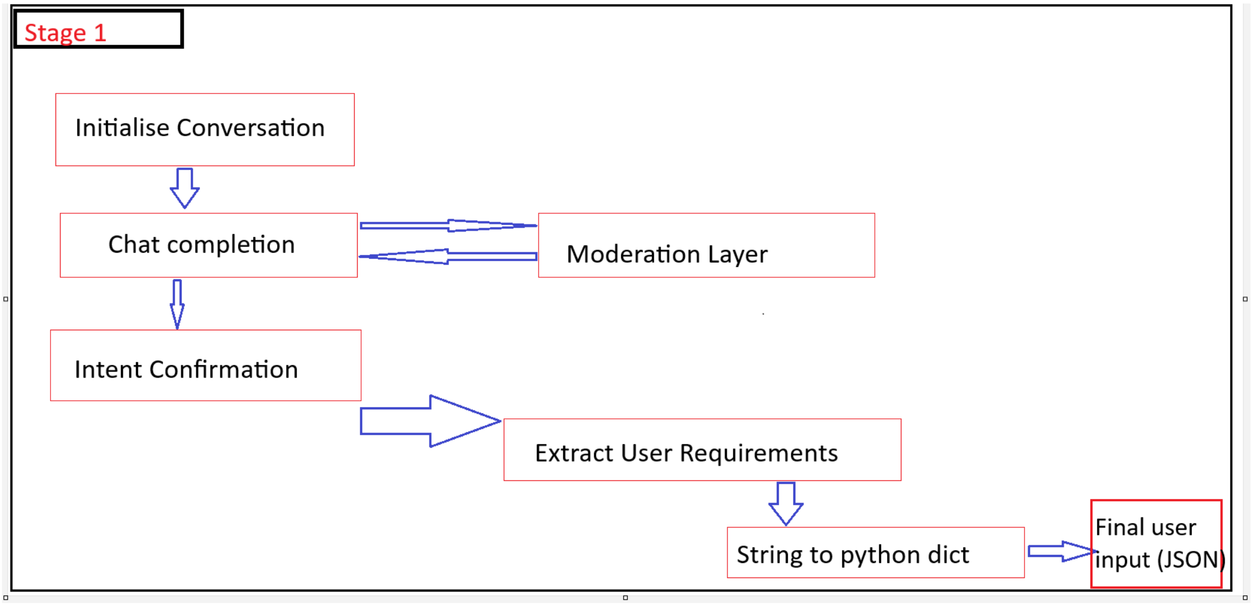
User will have some preference where to travel , how to travel , mode of travel , duration of travel , stay preference , budget .Taking all the points into consideration user requirement is gathered and then searched for the appropriate option in the DB .cased on the filter criteria the final result is fetched and given to user .For internal use , travel agency company has given package name to different package .At the end of the chat , when the final result is found based on the search criteria -user will be provided with the package name so that if user has to contact the travel agency company again he/she can directly give the reference using that package name .

Below is the csv file which is acting as DB of the travel agency company.

updated_dummy_travel_data.csv

Complete Project for the travel agency is divided into three stages

Stage 1 :



1. initilise_conversation

This method will be used to initialize the conversation, where you will describe a prompt stating that the system is a travel planner and you will provide on what all factors keeping into consideration the assistant has to respond back .

The response is stored in a variable and you can get the output for that by printing the variable.

Code :

```
import openai

import pandas as PD

import random


with open("/content/drive/MyDrive/Upgrad/APIKey.txt", "r") as f:

 openai.api_key = ' '.join(f.readlines())


def initilise_conversation():
 TravelData = {
     "travel budget":"100000",
     "travel motive":"Business",
     "travel type":"Solo",
     "travel destination":"Germany",
     "travel duration":"One month",
     "travel month":"December",
     "accommodation Preference":"5 star hotel",
     "travel mode" :"plane"
           }


 prompt = f'''You are a expert travel agent and has been working in travel industry from
last 10 years . You are considered as one of the best resource in suggesting the travel
package based on the user requirements .

 you have to ask them the requirement where the user wants visit whether the user
wishes to visit cold area or area having rainfall or desert area or high tech places which
are upto date in terms of technology or user want to visit country side

 .You have to confirm the budget and number of days for which user is planing to visit
.You should keep in mind to ask whether it is a solo trip or family trip .


 You may also consider asking about what type of stay user wants to have , like user
wants to stay in hotel or shared accomodation or user is going to stay at his own .Also
consider the travel mode into consideration ,whether user prefers to

 travel by train or plane or bus or taxi .


 you have to keep asking the questions from user till the time you donot have the
values for (travel budget , travel motive ,travel type , travel destination , travel duration ,
travel month) .All the keys (travel budget , travel motive ,travel type , travel destination ,
travel duration , travel month)

 should have data from the user based on the user requirement .
```

Also few points like visa requirement is there or not , whether it is a business travel or travelling with friends or with family .

Here is a sample conversation between the user and assistant:

User: "Hi, I am an travel buddy."

Assistant: "Great! As an traveller, you likely to travel somewhere and visit new place . Please share your requirement what is the motive of travel , travel duration and mode prefered for travelling and what is the maximum budget"

User: "I am travelling for my office meeting and it will be visit to canada"

Assistant: "Thank you for providing that information.Do you have some specific dates on which you prefer to travel "

User: "Yes, I would like to tavel in month of december near to new year"

Assistant: "Thank you for the information.I would like to ask whether you have visited canada earlier , do you hold valid visa for canada"

User: "Yes, I keep visiting canada and I have multiple entry visa with me "

Assistant:"Could you kindly let me know your budget for the travel? This will help me find options that fit within your price range while meeting the specified requirements."

User: "my max budget is 1.5lakh inr"

Assistant:"Are you flexible with date of travel"

User: "No , not at all .My travel dates are fixed .I will travel in december near to new year"

You can also refer to some chain of thoughts

####

You can refer to this chain of thought

Thought 1 : First thing when you start , try to understand the user requirement .What the user is looking for .Say if the user needs to travel for office work or business , then

```
    suggest a 5 star hotel , suggest user to travel via business class . if you understand the question , fill the respective value in the keys (travel budget , travel motive ,travel type , travel destination , travel duration , travel month)

    In case if you unable to understand the question , keep asking the question to user till the time you didnot understand the requirement completely

    and till the time you are unable to fill the values in (travel budget , travel motive ,travel type , travel destination , travel duration , travel month) based on customer requirement

  ####


  ####

    You can refer to this chain of thought

    Thought 1 : If the user wants to travel within same country (in which he is currently) then consider this as a domestic trip and suggest him a low budget user friend trip .

    Suggest him how he can save money by travelling in train instead of plane if money is a constraint .Suggest user to live in shared accomodation or stay in some paying guest .

  ####


  Start with a short welcome message and encourage the user to share their requirements.you have to keep asking the questions till the time you didnot get the answers to all the keys in the input {TravelData}

    '''


  messages_history = [{"role": "system", "content": prompt}]

  return messages_history
```

Method 2 : getChatCompletion

In this method , we are passing the above prompt as a message and calling the chat gpt chat completion api . This method will return a welcome messge from assistant , where the user will be asked how the travel agent can help him

Code :

```
def getChatCompletion(messages):
  chat_response = openai.chat.completions.create(
    model="gpt-4o",
    max_tokens=  900,
    temperature= 0,
    messages=messages
  )
  return chat_response.choices[0].message.content


debug_introduction = getChatCompletion(debug_conversation)
debug_introduction
```

Method 3 : ModerationCheck

This method will use the moderation api to check if the user has passed the valid /acceptable message or not . The **Moderation API** in chat completions is a tool designed to ensure that generated content adheres to predefined safety, ethical, and compliance guidelines. Here’s how it is used in the context of chat completions:

1. Content Filtering

- It evaluates the user’s input or the AI's response for potentially harmful, offensive, or inappropriate content.
- This includes identifying hate speech, harassment, violence, self-harm, explicit content, or any other policy-violating material.

2. Safety Assurance

- Prevents the system from producing or amplifying harmful outputs that could lead to misuse or real-world consequences.
- For example, filtering content that may encourage illegal activities or spread misinformation.

3. Ethical AI Deployment

- Ensures AI interactions align with ethical standards and the platform's terms of service.
- Moderation APIs allow developers to create a safer and more inclusive environment for users.

4. Customizability and Granularity

- Developers can set different thresholds for moderation based on their use case, audience, or industry regulations.
- For instance, a kid-friendly chatbot may require stricter filters compared to a professional advisory assistant.

5. Real-Time Monitoring

- The Moderation API works in real-time, analyzing both inputs and outputs to flag or block inappropriate content immediately.

6. Feedback Loop for Improvement

- Detected moderation issues can serve as feedback for retraining AI models to improve safety over time.
- Helps in reducing false positives or negatives in moderation.

Code :

```
#Moderation API to check the user input
from openai import OpenAI
with open("/content/drive/MyDrive/Upgrad/APIKey.txt", "r") as f:
  openai.api_key = ' '.join(f.readlines())
openai.api_key
client = OpenAI(api_key=openai.api_key)

def ModerationCheck(input_msg):

  errors = {
          "hate": "Content that expresses, incites, or promotes hate based on race, gender, ethnicity, religion, nationality, sexual orientation, disability status, or caste.",
          "hate/threatening": "Hateful content that also includes violence or serious harm towards the targeted group.",
          "self-harm": "Content that promotes, encourages, or depicts acts of self-harm, such as suicide, cutting, and eating disorders.",
          "sexual": "Content meant to arouse sexual excitement, such as the description of sexual activity, or that promotes sexual services (excluding sex education and wellness).",
          "sexual/minors": "Sexual content that includes an individual who is under 18 years old.",
          "violence": "Content that promotes or glorifies violence or celebrates the suffering or humiliation of others.",
          "violence/graphic": "Violent content that depicts death, violence, or serious physical injury in extreme graphic detail.",
      }


  client = OpenAI(api_key=openai.api_key)
  moderation_response=client.moderations.create(input=input_msg)


  categories=moderation_response.results[0].categories


  result=[]
  for category, value in categories:
    if value==True:
        result.append(errors[category])
  print(result)
```

Method 4 : intent_confirmation_layer

This method will check if the system has gathered all the required information from the user or not .

If yes then it will proceed further , if no then user will give the more input and assistant will then reply and again check and this loop keeps on working till the point , data for following keys: 'travel budget' , 'travel motive' ,'travel type' , 'travel destination' , 'travel duration' , 'travel month' is not gathered .

Code :

```
#This will help to check if system has all the parameters required from the user or not


from openai import OpenAI

with open("/content/drive/MyDrive/Upgrad/APIKey.txt", "r") as f:

  openai.api_key = ' '.join(f.readlines())

openai.api_key

client = OpenAI(api_key=openai.api_key)


def intent_confirmation_layer(response_assistant):

    prompt = """

    You are a senior evaluator who has an eye for detail.

    You are provided an input. You need to evaluate if the input has the following keys: 'travel budget' , 'travel motive' ,'travel type' , 'travel destination' , 'travel duration' , 'travel month'

    Next you need to evaluate if the keys have the the values filled correctly.

    The values for all keys, except 'budget', should be 'low', 'medium', or 'high' based on the importance as stated by user.

    The value for the key 'budget' needs to contain a number with currency.

    Output a string 'Yes' if the input contains the dictionary with the values correctly filled for all keys.

    Otherwise out the string 'No'.


    Here is the input: {debug_response_assitant}
    Only output a one-word string - Yes/No.
    """


    confirmation = client.completions.create(
                                model="gpt-3.5-turbo-instruct",
                                prompt = prompt,
                                temperature=0)




    return confirmation.choices[0].text
```

Method 5 : extract_userDemands

This method will provide the user requirement in form of json which is not in json format in actual .but the type is string and it some how looks as json .This method will provide the output like {

```
  "Travel Motive": "Solo trip",
  "Travel Destination": "Canada",
  "Travel Duration": "10 days",
  "Travel Month": "Feb",
  "Travel Budget": "5000000",
  "Accommodation Preference": "3-star hotel",
  "Travel Mode": "Business class plane"
}
```

The above format seems like it is in json but actually its type is str .

Code :

```
#This will help to extract the key value pairs from the complete response
```

```
from openai import OpenAI
with open("/content/drive/MyDrive/Upgrad/APIKey.txt", "r") as f:
  openai.api_key = ' '.join(f.readlines())
openai.api_key
client = OpenAI(api_key=openai.api_key)


def extract_userDemands(response_assistant):
    prompt = """
    You are tasked with extracting key-value pairs from a provided text, which summarizes user information in a structured format. Your goal is to identify the keys and their corresponding values accurately and output them as a JSON object.


Instructions:
Look for the section in the text that outlines the details provided by the user. This will typically appear as bullet points or a list of attributes and their values. Ignore any additional questions or commentary outside this section.


Extract:


Keys: The bolded or labeled attributes (e.g., Travel Motive, ABCD, etc.).
Values: The corresponding information associated with each key (e.g., Solo trip, QWERTY, etc.).
Format the output as a JSON object with:


Each extracted key as a JSON property.
Each corresponding value as the property's value.
Ensure proper formatting, with keys as strings and values as strings or appropriate formats (e.g., numbers without commas).
```

Remove any units or formatting from numerical values (e.g., "5,000,000 INR" → "5000000").

Handle empty or missing values by assigning an empty string ("") as the value.

Retain proper capitalization and spacing as presented in the original text.

Example Input and Output:

Input Example 1: Thank you for sharing your requirements. Here's a summary of the information you've provided:

Travel Motive: Solo trip

Travel Destination: Canada

Travel Duration: 10 days

Travel Month: Feb

Travel Budget: 5,000,000 INR

Accommodation Preference: 3-star hotel

Travel Mode: Business class plane

Could you please confirm if you have a valid visa for Canada, or if you need assistance with the visa application process? Additionally, are there any specific activities or places in Canada you are particularly interested in visiting during your stay?

Output Example 1:

```
{
  "Travel Motive": "Solo trip",
  "Travel Destination": "Canada",
  "Travel Duration": "10 days",
  "Travel Month": "Feb",
  "Travel Budget": "5000000",
  "Accommodation Preference": "3-star hotel",
  "Travel Mode": "Business class plane"
}
```

Input Example 2: Thank you for sharing your requirements. Here's a summary of the information you've provided:

ABCD: QWERTY

IFGH: PARIS

ASDF: 25 months

TYUI: September

POIUY: 599 USD

UIOP: UJHG

CCVV: Business

Could you please confirm if you have a valid visa for Canada, or if you need assistance with the visa application process? Additionally, are there any specific activities or places in Canada you are particularly interested in visiting during your stay?

Output Example 2:

```
{
  "ABCD": "QWERTY",
  "IFGH": "PARIS",
  "ASDF": "25 months",
  "TYUI": "September",
  "POIUY": "599",
  "UIOP": "UJHG",
  "CCVV": "Business"
}
```

```
Provide a text input to extract the key-value pairs, and ensure the output strictly follows the JSON format described above.

    Here is the input: {debug_response_assitant}

    """


    confirmation = client.completions.create(

                                model="gpt-3.5-turbo-instruct",

                                prompt = prompt,

                                temperature=0,

                                max_tokens=900)




    return confirmation.choices[0].text
```

Method 6 : extract_dictionary_from_string

This method will convert the previous method output which was in form of str to json .It will be actually in json form now . This json will have all the keys and values . The output will have the following keys: 'travel budget' , 'travel motive' ,'travel type' , 'travel destination' , 'travel duration' , 'travel month' and its corresponding values .

Code :

```
#Code below will help to convert the String input to python dict

import ast

import re


def extract_dictionary_from_string(string):

    regex_pattern = r"\{[^{}]+\}"


    dictionary_matches = re.findall(regex_pattern, string)
    # Extract the first dictionary match and convert it to lowercase
    if dictionary_matches:
        dictionary_string = dictionary_matches[0]
        dictionary_string = dictionary_string.lower()

        # Convert the dictionary string to a dictionary object using ast.literal_eval()
        dictionary = ast.literal_eval(dictionary_string)
    return dictionary
```

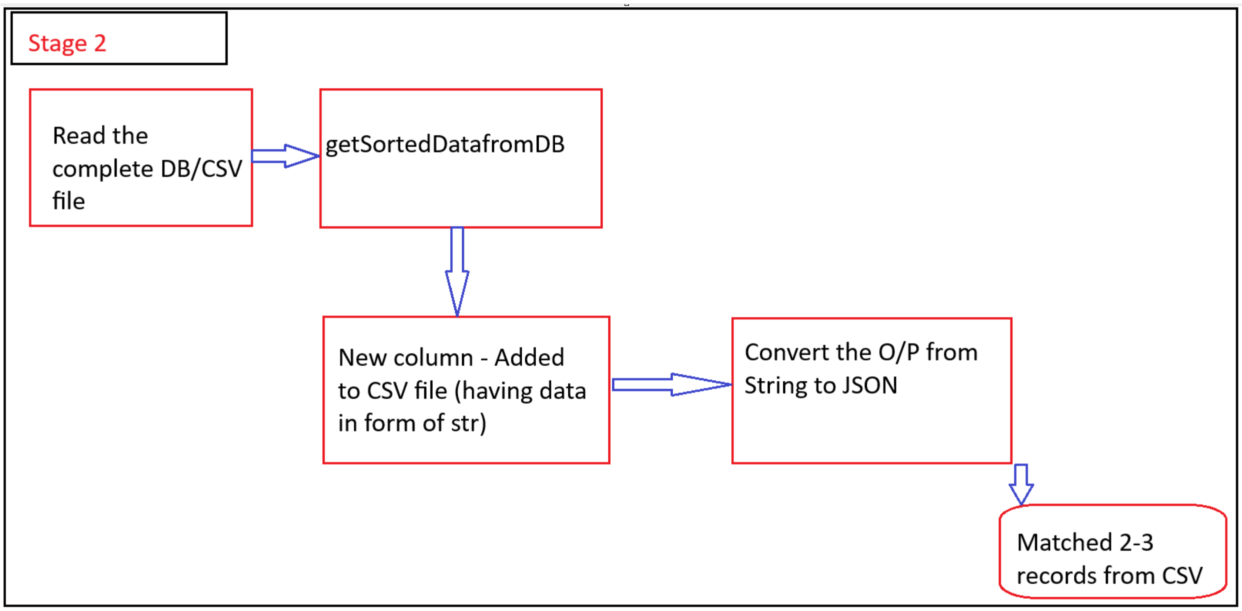
Here you will have the final user input (what user wants) and that too in form of JSON

STAGE 2 :

In this stage , we will read the data from DB .here as a DB we have CSV file (attached above) which has the data for Travel type,PackageName,Travel Destination,Travel Duration,Travel Budget,Accommodation,Travel Mode,Recommended month of travel,On arrival visa,Extra Taxes on stay,Free Meals,Free Airport pick and drop available,Hotels available,Nearby attractions,Pets allowed in hotel,Swimming pools in hotel,Free amenities in hotel,Summary.

Alike every company has some internal data , this is a travel agency internal data .Every sort of entry in the sheet will have a unique package name for internal reference and the system will provide the package name to customer end of the day so that if that customer want to chat again with reference to package name ,it will be minimal efforts from the customer and chatbot side .

We will first read the complete csv file and then store the result in new column , in form of json (which will actually look like json but is str in actual).Again here the moderation layer will be used to check if the output from assistant is safe and it doesnot have any sort of irrelevant data which can be considered as threat and not safe .


Stage 2
Read the complete DB/CSV file
getSortedDatafromDB
New column - Added to CSV file (having data in form of str)
Convert the O/P from String to JSON
Matched 2-3 records from CSV

Method 1 : getSortedDatafromDB

This method will first read the complete csv file and give the data , There is a column in csv file (Column : Summary) .This summary column has details of all the destination why the destination is famous , what all hotels are there , budget , travel mode , famous places etc

Example : Switzerland is a great place to visit, offering diverse attractions and activities. Make sure to plan your trip to enjoy the best of what it offers.Enjoy the beautiful market places , cafes , scenaric beauty with family and friends .You may also involve in plethora of adventurous activies and can plan trip of around 8 to 15 days and costing somewhat around 600000 INR to 900000 INR .Enjoy the pool sode room with window facing towards sea and enjoy the tasty food and drinks with complementary visit to near by market by hotel shuttle .Recommend to plan trip in November or December and for budget trip plan a trip by economic class plane

Code :

```
#Stage 2 : you will create a dummy data having say 10-15 records , you will read that csv file

import pandas as PD

import numpy as NP

import openai

with open("/content/drive/MyDrive/Upgrad/APIKey.txt", "r") as f:
  openai.api_key = ' '.join(f.readlines())

openai.api_key

client = openai.OpenAI(api_key=openai.api_key)



#Read the Csv file for laptop data

PDD = PD.read_csv("/content/drive/MyDrive/Upgrad/updated_dummy_travel_data.csv")

Travel_Details = PDD['Summary']

Travel_Details


def getSortedDatafromDB(input):


  prompt = f"""you are a excellent evaluator and has good experience in sorting the data based on the required format .

  you have to read the input in form of plain english language and extract the mentioned key and values out of that



    while preparing the data you have to keep the following point into consideration :

    Here in the above json format  travel motive can have value Business trip or solo trip or Family vacation or Adventure trip or Honeymoon   .

    Here in the above json format  travel destination can have values like Canada  or Australia  or Switzerland  or Japan  or Thailand  or France  or Italy  or Maldives  or Dubai  or South Africa  or Brazil  or New Zealand  or Spain  or Singapore  or USA  or Mexico  or India  or Germany  or Greece  or Iceland  or Canada  Vietnam  or Turkey  or Norway  or  Indonesia  or Argentina

    Here in the above json format  travel duration can have values in days like 10 days or 15 days or 8 days or 40 days and so on

    Here in the above json format  travel month can have values in months like january ,febuary , march , and son on till december

    Here in the above json format  travel mode can have values like Business class plane or Economy class plane or First class plane
```

You have to prepare the data which should look like json having keys (travel motive , travel destination , travel duration , travel month , travel mode)

you can consider few shots mentioned below for reference :

Example 1 : Switzerland is a great place to visit, offering diverse attractions and activities.

Make sure to plan your trip to enjoy the best of what it offers.Enjoy the beautiful market places , cafes , scenaric beauty with family and friends .

You may also involve in plethora of adventurous activies make your trip as Adventurous trip and can plan trip of around 8 to 15 days and costing somewhat around 600000 INR to 900000 INR .

Enjoy the pool sode room with window facing towards sea and enjoy the tasty food and drinks with complementary visit to near by market by hotel shuttle .

Recommend to plan trip in novemeber or decemeber and for budget trip plan a trip by economic class plane .

Output :

```
  {{
  "Travel Motive": "Adventurous trip",
  "Travel Destination": "Switzerland",
  "Travel Duration": "8 days",
  "Travel Month": "November",
  "Accommodation Preference": "5 star hotel",
  "Travel Mode": "Economic class plane"
}}
```

Provide a text input to extract the key-value pairs, and ensure the output strictly follows the JSON format described above.

```
   Here is the input: {Travel_Details}
   """


 confirmation = client.completions.create(
                          model="gpt-3.5-turbo-instruct",
                          prompt = prompt,
                          temperature=0,
                          max_tokens=900)



 return confirmation.choices[0].text
```

METHOD 2 : addDetailsInJSONFormat

Here we will read this summary column and read its data and will then extract the keys from it and its value so that we can get the json like object out of it .

Expected output : \n{\n "Travel Motive": "Family vacation",\n "Travel Destination": "Canada",\n "Travel Duration": "10 days",\n "Travel Month": "February",\n "Accommodation Preference": "5 star hotel",\n "Travel Mode": "Business class plane"\n}

Now we will this output for all destinations in new column in csv file so that we can have this sort of output / data in that column for all destinations

Code :

```
SortedDataFromDBSource = getSortedDatafromDB(Travel_Details)

SortedDataFromDBSource

PDD['Travel_Summary'] = PDD['Summary'].apply(lambda x: getSortedDatafromDB(x))

PDD.to_csv("/content/drive/MyDrive/Upgrad/updated1_dummy_travel_data.csv",index=False,header=True)
```

METHOD 3 : getPackageNameBasedOnUserReq

This method will help to get the package name based on all the user requirements. Package name – travel agency has set for internal use .First it will get what destination user wants to travel , then you will 2-3 options out of that .Further user will be asked budget , based on user budget it will filter the data and will provide the best suitable package for that destination within user budget

Code :

```
#Sorting the data from csv based on the user requirement


import numpy as NP
import pandas as PD
import re


##User Requirement
UserTravelRequirement


def getPackageNameBasedOnUserReq():
  userBudget = UserTravelRequirement['travel budget']
  userTravelDestination = (UserTravelRequirement['travel destination']).lower()


  #Lets create a copy of the laptop updated csv and then make the changes in that
  PDF= PDD.copy()


  # Convert 'Price' column to numeric, handling errors by coercing to NaN
  PTT = PDF['Travel Budget']
  # Convert pandas series to list of strings
  LL = PTT.astype(str).tolist()
  # Apply the filtering and conversion to each element of the list using a list comprehension
  priceValuesInCSVFile = [int(re.sub(r"[^0-9]", "", item)) for item in LL]
  priceValuesInCSVFile

  #Sorting the data based on destinationa and budget from the csv file
  PTT = PDF[PDF['Travel Destination'].str.lower().isin([userTravelDestination])]
  FilteredRecordFromCSVOnUserPref = PTT[PTT['Travel Budget']<=userBudget]
  FilteredRecordFromCSVOnUserPref
  return FilteredRecordFromCSVOnUserPref['PackageName']

 SuggestedTravelPackage = getPackageNameBasedOnUserReq()
SuggestedTravelPackage
```