```
from datetime import datetime, timedelta
from data_manager import DataManager
from flight_search import FlightSearch
from notification_manager import NotificationManager

data_manager = DataManager()
sheet_data = data_manager.get_destination_data()
flight_search = FlightSearch()
notification_manager = NotificationManager()

ORIGIN_CITY_IATA = "LON"

if sheet_data[0]["iataCode"] == "":
    for row in sheet_data:
        row["iataCode"] = flight_search.get_destination_code(row["city"])
    data_manager.destination_data = sheet_data
    data_manager.update_destination_codes()

tomorrow = datetime.now() + timedelta(days=1)
six_month_from_today = datetime.now() + timedelta(days=(6 * 30))

for destination in sheet_data:
    flight = flight_search.check_flights(
        ORIGIN_CITY_IATA,
        destination["iataCode"],
        from_time=tomorrow,
        to_time=six_month_from_today
    )
    if flight.price < destination["lowestPrice"]:
        notification_manager.send_sms(
            message=f"Low price alert! Only £{flight.price} to fly from {flight.origin_city}-{flight.origin_airport} to {flight.destination_city}-{flight.destination_airport}, from {flight.out_date} to {flight.return_date}."
        )


#flight_data
```

```
36 class FlightData:
37
38     def __init__(self, price, origin_city,
   origin_airport, destination_city, destination_airport
   , out_date, return_date):
39         self.price = price
40         self.origin_city = origin_city
41         self.origin_airport = origin_airport
42         self.destination_city = destination_city
43         self.destination_airport =
   destination_airport
44         self.out_date = out_date
45         self.return_date = return_date
46
47 #data_manager
48 from pprint import pprint
49 import requests
50
51 SHEETY_PRICES_ENDPOINT = "https://api.sheety.co/***/
   flightDeals/prices"
52
53
54 class DataManager:
55
56     def __init__(self):
57         self.destination_data = {}
58
59     def get_destination_data(self):
60         response = requests.get(url=
   SHEETY_PRICES_ENDPOINT)
61         data = response.json()
62         self.destination_data = data["prices"]
63         return self.destination_data
64
65     def update_destination_codes(self):
66         for city in self.destination_data:
67             new_data = {
68                 "price": {
69                     "iataCode": city["iataCode"]
70                 }
71             }
```

```
            response = requests.put(
                url=f"{SHEETY_PRICES_ENDPOINT}/{city['id']}",
                json=new_data
            )
            print(response.text)

#Notification_manager
from twilio.rest import Client

TWILIO_SID = "**************"
TWILIO_AUTH_TOKEN = "************"
TWILIO_VIRTUAL_NUMBER = "********"
TWILIO_VERIFIED_NUMBER = "*******"


class NotificationManager:

    def __init__(self):
        self.client = Client(TWILIO_SID, TWILIO_AUTH_TOKEN)

    def send_sms(self, message):
        message = self.client.messages.create(
            body=message,
            from_=TWILIO_VIRTUAL_NUMBER,
            to=TWILIO_VERIFIED_NUMBER,
        )
        # Prints if successfully sent.
        print(message.sid)

#flight_search
import requests
from flight_data import FlightData

TEQUILA_ENDPOINT = "https://tequila-api.kiwi.com"
TEQUILA_API_KEY = "*********"


class FlightSearch:

```

```
111     def get_destination_code(self, city_name):
112         location_endpoint = f"{TEQUILA_ENDPOINT}/
    locations/query"
113         headers = {"apikey": TEQUILA_API_KEY}
114         query = {"term": city_name, "location_types"
    : "city"}
115         response = requests.get(url=
    location_endpoint, headers=headers, params=query)
116         results = response.json()["locations"]
117         code = results[0]["code"]
118         return code
119 
120     def check_flights(self, origin_city_code,
    destination_city_code, from_time, to_time):
121         headers = {"apikey": TEQUILA_API_KEY}
122         query = {
123             "fly_from": origin_city_code,
124             "fly_to": destination_city_code,
125             "date_from": from_time.strftime("%d/%m/%
    Y"),
126             "date_to": to_time.strftime("%d/%m/%Y"),
127             "nights_in_dst_from": 7,
128             "nights_in_dst_to": 28,
129             "flight_type": "round",
130             "one_for_city": 1,
131             "max_stopovers": 0,
132             "curr": "GBP"
133         }
134 
135         response = requests.get(
136             url=f"{TEQUILA_ENDPOINT}/v2/search",
137             headers=headers,
138             params=query,
139         )
140 
141         try:
142             data = response.json()["data"][0]
143         except IndexError:
144             print(f"No flights found for {
    destination_city_code}.")
145             return None
```

```
146
147         flight_data = FlightData(
148             price=data["price"],
149             origin_city=data["route"][0]["cityFrom"
    ],
150             origin_airport=data["route"][0]["flyFrom
    "],
151             destination_city=data["route"][0]["
    cityTo"],
152             destination_airport=data["route"][0]["
    flyTo"],
153             out_date=data["route"][0]["
    local_departure"].split("T")[0],
154             return_date=data["route"][1]["
    local_departure"].split("T")[0]
155         )
156         print(f"{flight_data.destination_city}: £{
    flight_data.price}")
157         return flight_data
158
159
```