class PetrolStationMaintenance:

def \_\_init\_\_(self):

self.records = []

def add\_record(self, station\_id, date, task):

# Add a new maintenance record

self.records.append({'station\_id': station\_id, 'date': date, 'task': task})

print("Record added successfully.")

def view\_records(self):

# Display all maintenance records

for record in self.records:

print(f"Station ID: {record['station\_id']}, Date: {record['date']}, Task: {record['task']}")

def update\_record(self, station\_id, date, new\_task):

# Update an existing maintenance record

for record in self.records:

if record['station\_id'] == station\_id and record['date'] == date:

record['task'] = new\_task

print("Record updated successfully.")

return

print("Record not found.")

def delete\_record(self, station\_id, date):

# Delete a maintenance record

self.records = [record for record in self.records if not (record['station\_id'] == station\_id and record['date'] == date)]

print("Record deleted successfully.")

if \_\_name\_\_ == "\_\_main\_\_":

maintenance\_tracker = PetrolStationMaintenance()

station\_id = input("Enter Station ID: ")

date = input("Enter Date (YYYY-MM-DD): ")

task = input("Enter Task Description: ")

maintenance\_tracker.add\_record(station\_id, date, task

maintenance\_tracker.view\_records()