Part-1 Jothiswaran.P

Q1. Prime Check  
def is\_prime(n):  
 if n <= 1:  
 return False  
 for i in range(2, int(n\*\*0.5) + 1):  
 if n % i == 0:  
 return False  
 return True

print(is\_prime(7))

Q2. Reverse & Palindrome Check  
s = input("Enter a string: ")  
rev = s[::-1]  
print("Reversed:", rev)  
print("Palindrome?", s.lower() == rev.lower())

Q3.  
nums = [5, 2, 9, 5, 2, 8]  
nums = sorted(set(nums))  
print("Second largest:", nums[-2])  
Part-2

Q4.   
class Person:  
 def \_\_init\_\_(self, name, age):  
 self.name = name  
 self.age = age  
 def display(self):  
 print(f"Name: {self.name}, Age: {self.age}")  
  
class Employee(Person):  
 def \_\_init\_\_(self, name, age, employee\_id, department):  
 super().\_\_init\_\_(name, age)  
 self.employee\_id = employee\_id  
 self.department = department  
 def display(self):  
 print(f"Name: {self.name}, Age: {self.age}, ID: {self.employee\_id}, Department: {self.department}")  
  
e = Employee("Jo", 48, "E101", "DE")  
e.display()

Q5. Method Overriding   
class Vehicle:  
 def drive(self):  
 print("Vehicle is driving")  
  
class Car(Vehicle):  
 def drive(self):  
 print("Car is driving smoothly")  
  
c = Car()  
c.drive()

Part-3

Q6. Clean CSV  
import pandas as pd  
import numpy as np  
  
df = pd.read\_csv("students.csv")  
df['Age'].fillna(df['Age'].mean())  
df['Score'].fillna(0)  
df.to\_csv("students\_cleaned.csv", index=False)  
print(df)  
Q7. Convert Cleaned CSV to JSON  
df = pd.read\_csv("students\_cleaned.csv")  
df.to\_json("students.json", orient="records", indent=4)

Part-4

Q8.   
df = pd.read\_csv("students\_cleaned.csv")  
def get\_status(score):  
 if score >= 85:  
 return "Distinction"  
 elif score >= 60:  
 return "Passed"  
 else:  
 return "Failed"  
  
df['Status'] = df['Score'].apply(get\_status)  
df['Tax\_ID'] = "TAX-" + df['ID'].astype(str)  
df.to\_csv("students\_cleaned.csv", index=False)

print(df)

Part-5

Q9. Increase Prices by 10%

import json  
  
with open("products.json") as f:  
 products = json.load(f)  
  
for p in products:  
 p['price'] = round(p['price'] \* 1.1, 2)  
  
with open("products\_updated.json", "w") as f:  
 json.dump(products, f, indent=4)  
  
print(products)