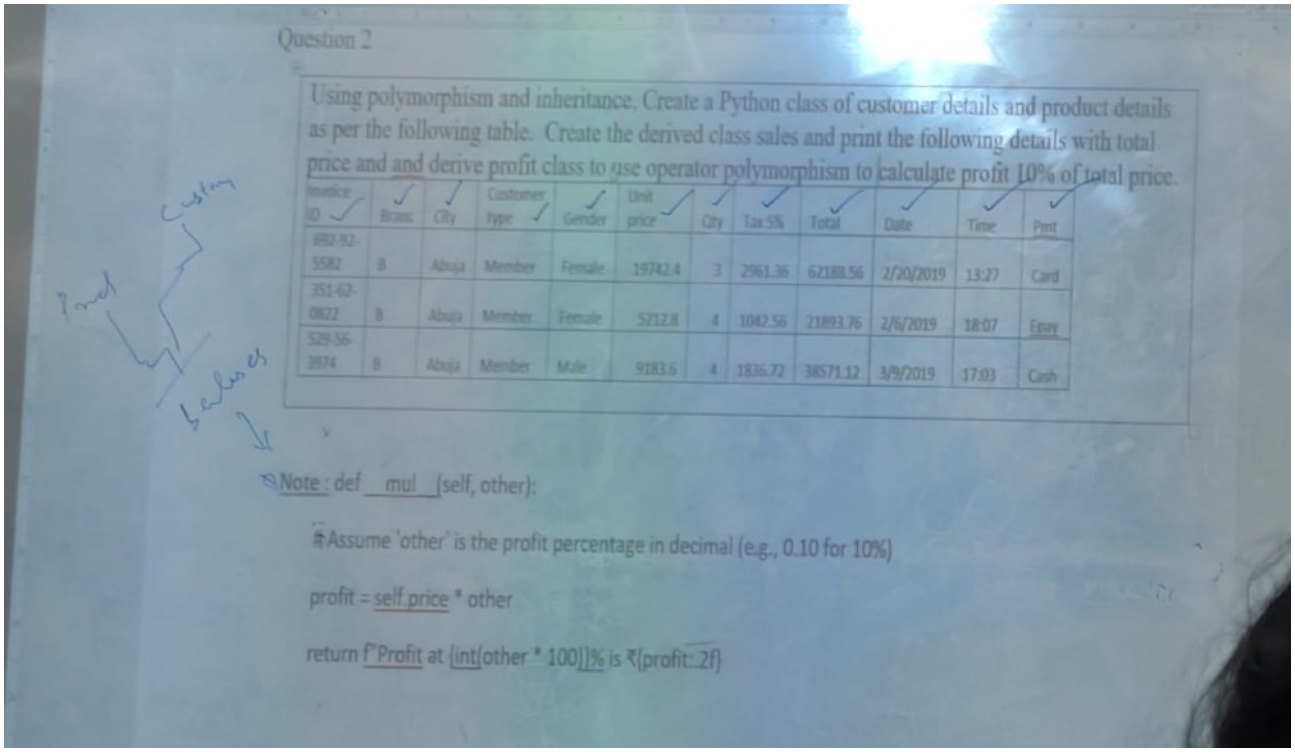

Question 2

Using polymorphism and inheritance, Create a Python class of customer details and product details as per the following table. Create the derived class sales and print the following details with total price and and derive profit class to use operator polymorphism to calculate profit 10% of total price.

Invoice ID	Branc	City	Customer type	Gender	Unit price	Qty	Tax 5%	Total	Date	Time	Pmt
692-92-5582	B	Abuja	Member	Female	19742.4	3	2961.36	62188.56	2/20/2019	13:27	Card
351-62-0822	B	Abuja	Member	Female	5212.8	4	1042.56	21893.76	2/6/2019	18:07	Epay
529-56-3974	B	Abuja	Member	Male	9183.6	4	1836.72	38571.12	3/9/2019	17:03	Cash

Note : def __mul__(self, other):

#Assume 'other' is the profit percentage in decimal (e.g., 0.10 for 10%)

profit = self.price * other

return f"Profit at {int(other * 100)}% is ₹{profit:.2f}

Python Classes for Customer, Product, Sales, and Profit Calculation

You are asked to use:

- **Inheritance** (base & derived classes)
- **Polymorphism** (operator overloading)
- The table fields for customer and sales details.
- Implement the formula: “Profit = 10% of total price" using operator overloading.

1. Define Base Classes: `CustomerDetails`, `ProductDetails`

```
python
class CustomerDetails:
    def __init__(self, city, customer_type, gender):
        self.city = city
        self.customer_type = customer_type
        self.gender = gender

class ProductDetails:
    def __init__(self, branch, unit_price, quantity, tax,
total, date, time, payment):
```

```
        self.branch = branch
        self.unit_price = unit_price
        self.quantity = quantity
        self.tax = tax
        self.total = total
        self.date = date
        self.time = time
        self.payment = payment
```

2. Derived Class: `Sales`

```
python
class Sales(CustomerDetails, ProductDetails):
    def __init__(self, invoice_id, branch, city,
customer_type, gender, unit_price, quantity, tax, total,
date, time, payment):
        CustomerDetails.__init__(self, city, customer_type,
gender)
        ProductDetails.__init__(self, branch, unit_price,
quantity, tax, total, date, time, payment)
        self.invoice_id = invoice_id

    def show_details(self):
        print(f"Invoice ID: {self.invoice_id} | Branch:
{self.branch} | City: {self.city} | Customer Type:
{self.customer_type} | Gender: {self.gender} | Unit Price:
{self.unit_price} | Quantity: {self.quantity} | Tax:
{self.tax} | Total: {self.total} | Date: {self.date} | Time:
{self.time} | Payment: {self.payment}")
```

3. `Profit` class with Operator Overloading

```
python
class Profit:
    def __init__(self, price):
        self.price = price

    def __mul__(self, other):  # other is the profit percent
in decimal, e.g., 0.1 for 10%
        profit = self.price * other
        return f"Profit at {int(other*100)}% is ₹
{profit:.2f}"
```

Full Example with Data from Table

python

```python
# Create a sales object from the first row of your table
sale1 = Sales(
    invoice_id="692-52", branch="B", city="Abuja",
customer_type="Member", gender="Female",
    unit_price=19742.4, quantity=3, tax=2961.36,
total=62188.56,
    date="2/20/2019", time="13:27", payment="Card"
)

sale2 = Sales(
    invoice_id="5582", branch="B", city="Abuja",
customer_type="Member", gender="Female",
    unit_price=5212.8, quantity=4, tax=1042.56,
total=21983.76,
    date="2/6/2019", time="18:07", payment="Epay"
)

sale3 = Sales(
    invoice_id="351-62", branch="B", city="Abuja",
customer_type="Member", gender="Male",
    unit_price=9183.6, quantity=4, tax=1836.72,
total=38571.12,
    date="3/9/2019", time="17:03", payment="Cash"
)

# Print details
sale1.show_details()
sale2.show_details()
sale3.show_details()

# Calculate profit using operator overloading (10% = 0.10)
profit1 = Profit(sale1.total)
profit2 = Profit(sale2.total)
profit3 = Profit(sale3.total)

print(profit1 * 0.10)
print(profit2 * 0.10)
print(profit3 * 0.10)
```

Output Example

text

```
Invoice ID: 692-52 | Branch: B | City: Abuja | Customer Type:
Member | Gender: Female | Unit Price: 19742.4 | Quantity: 3 |
Tax: 2961.36 | Total: 62188.56 | Date: 2/20/2019 | Time:
13:27 | Payment: Card
Invoice ID: 5582 | Branch: B | City: Abuja | Customer Type:
Member | Gender: Female | Unit Price: 5212.8 | Quantity: 4 |
Tax: 1042.56 | Total: 21983.76 | Date: 2/6/2019 | Time: 18:07
| Payment: Epay
Invoice ID: 351-62 | Branch: B | City: Abuja | Customer Type:
Member | Gender: Male | Unit Price: 9183.6 | Quantity: 4 |
Tax: 1836.72 | Total: 38571.12 | Date: 3/9/2019 | Time: 17:03
| Payment: Cash
Profit at 10% is ₹6218.86
Profit at 10% is ₹2198.38
Profit at 10% is ₹3857.11
```

Summary

- **Inheritance**: Sales class inherits customer and product details.
- **Polymorphism**: Profit class overloads the `*` operator to compute profit.
- **Usage**: You create sales objects, print details, and calculate profits as in the table.