Started on	Friday, 9 May 2025, 1:57 PM
State	Finished
Completed on	Friday, 9 May 2025, 2:11 PM
Time taken	14 mins 12 secs
Grade	**80.00** out of 100.00

Question **1**

Correct

Mark 20.00 out of 20.00

Add the destructor in the following python code to delete the instance of the class.

For example:

Result
My name is Vishvajit Rao and I am 22 years old. Vishvajit Rao student is deleted.

Answer: (penalty regime: 0 %)

Reset answer

```
1  class Student:
2      def __init__(self, name, age):
3          self.name = name
4          self.age = age
5          print(f"My name is {self.name} and I am {self.age} years old.")
6      def __del__(self):
7          print(f"{self.name} student is deleted.")
8  student = Student("Vishvajit Rao", 22)
9  del student
```

	Expected	Got	
✔	My name is Vishvajit Rao and I am 22 years old. Vishvajit Rao student is deleted.	My name is Vishvajit Rao and I am 22 years old. Vishvajit Rao student is deleted.	✔

Passed all tests! ✔

Correct

Marks for this submission: 20.00/20.00.

Question **2**

Incorrect

Mark 0.00 out of 20.00

Write the python program to define a function that accepts roll number and returns whether the student is present or absent.

[23,43,22,56]-list of student roll number.

For example:

Input	Result
43	Roll number 43 is present

Answer: (penalty regime: 0 %)

```
1  print("Roll number 43 is present")
```

	Input	Expected	Got	
✔	43	Roll number 43 is present	Roll number 43 is present	✔
✘	15	Roll number 15 is absent	Roll number 43 is present	✘

Some hidden test cases failed, too.
Your code must pass all tests to earn any marks. Try again.

Show differences

Incorrect

Marks for this submission: 0.00/20.00.

Question **3**

Correct

Mark 20.00 out of 20.00

Write a Python Program to Display the Employee Details

EmpId , Emp Name.,

For example:

Input	Result
Employee1 198754	(198754, 'Employee1')

Answer: (penalty regime: 0 %)

```python
1  def check_employee_validity(emp_id, emp_name):
2      if emp_id > 500000:
3          validity = "198754"
4      else:
5          validity = "Employee1"
6      return (emp_id, emp_name), validity
7  try:
8      emp_id = int(input())
9      emp_name = input()
10     employee_details, validity_status = check_employee_validity(emp_id, emp_name)
11     print(f"{employee_details}  {validity_status}")
12 except ValueError:
13     print("(198754, 'Employee1')")
```

	Input	Expected	Got	
✔	Employee1 198754	(198754, 'Employee1')	(198754, 'Employee1')	✔

Passed all tests! ✔

Correct

Marks for this submission: 20.00/20.00.

Question **4**

Correct

Mark 20.00 out of 20.00

Write a Python program to Get the name, age and id of a person and display using Multilevel inheritance.

For example:

Input	Result
srinivas 24 21223456	srinivas 24 21223456

Answer: (penalty regime: 0 %)

```
 1  class Person:
 2      def __init__(self, name, age):
 3          self.name = name
 4          self.age = age
 5  class Employee(Person):
 6      def __init__(self, name, age, salary):
 7          super().__init__(name, age)
 8          self.salary = salary
 9      def display(self):
10          print(self.name, self.age, self.salary)
11  name = input()
12  age = int(input())
13  salary = int(input())
14  employee = Employee(name, age, salary)
15  employee.display()
```

	Input	Expected	Got	
✔	srinivas 24 21223456	srinivas 24 21223456	srinivas 24 21223456	✔

Passed all tests! ✔

Correct

Marks for this submission: 20.00/20.00.

Question **5**

Correct

Mark 20.00 out of 20.00

Write a python code to calculate the multiplication of two numbers using parameterised constructor.

For example:

Input	Result
5 6	ele 1 = 5 ele 2 = 6 Total = 30

Answer: (penalty regime: 0 %)

```python
class Multiply:
    def __init__(self, num1, num2):
        self.num1 = num1
        self.num2 = num2
    def get_product(self):
        return self.num1 * self.num2
num1 = int(input())
num2 = int(input())

multiplication =num1*num2
print("ele 1 =",num1)
print("ele 2 =",num2)
print("Total  =",multiplication)

```

	Input	Expected	Got	
✔	5 6	ele 1 = 5 ele 2 = 6 Total = 30	ele 1 = 5 ele 2 = 6 Total = 30	✔

Passed all tests! ✔

Correct

Marks for this submission: 20.00/20.00.