Started on	Saturday, 19 April 2025, 2:12 PM
State	Finished
Completed on	Saturday, 19 April 2025, 2:44 PM
Time taken	31 mins 55 secs
Grade	**80.00** out of 100.00

Question **1**

Correct

Mark 20.00 out of 20.00

Write a Python class named Student with two attributes student_id, student_name, get the values from the user. Create a function to display the entire attribute and their values in Student class.

For example:

Input	Result
V10 Jacqueline Barnett	Original attributes and their values of the Student class: Student id: V10 Student Name: Jacqueline Barnett

Answer: (penalty regime: 0 %)

```
class Student:
    def __init__(self , student_id, student_name):
        self.student_id = student_id
        self.student_name = student_name
    def display(self):
        print("Original attributes and their values of the Student class:")
        print("Student id:", self.student_id)
        print("Student Name:",self.student_name)
sid = input()
sname = input()
student = Student(sid, sname)
student.display()
```

	Input	Expected	Got	
✔	V10 Jacqueline Barnett	Original attributes and their values of the Student class: Student id: V10 Student Name: Jacqueline Barnett	Original attributes and their values of the Student class: Student id: V10 Student Name: Jacqueline Barnett	✔
✔	M19 Joey Tribbiani	Original attributes and their values of the Student class: Student id: M19 Student Name: Joey Tribbiani	Original attributes and their values of the Student class: Student id: M19 Student Name: Joey Tribbiani	✔

Passed all tests! ✔

Correct

Marks for this submission: 20.00/20.00.

Question **2**

Correct

Mark 20.00 out of 20.00

.get() is not a list method. Place **pass** keyword to the right line so that program doesn't throw an error.

For example:

Result
[1, 3, 5]

Answer: (penalty regime: 0 %)

Reset answer

```
#Type your code here.

a = [1, 3, 5]
try:
    a.get()
except:
    pass


print(a)

```

	Expected	Got	
✔	[1, 3, 5]	[1, 3, 5]	✔

Passed all tests! ✔

Correct

Marks for this submission: 20.00/20.00.

Question **3**

Correct

Mark 20.00 out of 20.00

Write a Python function that takes a list and returns a new list with unique elements of the first list.

unique_list=[1,2,3,3,3,3,4,5]

For example:

Result
[1, 2, 3, 4, 5]

Answer: (penalty regime: 0 %)

```
def get_unique_elements(lst):
    unique = []
    for item in lst:
        if item not in unique:
            unique.append(item)
    return unique
unique_list =[1, 2, 3, 4, 5]
print(get_unique_elements(unique_list))
```

	Expected	Got	
✔	[1, 2, 3, 4, 5]	[1, 2, 3, 4, 5]	✔

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 perform multiplication and floor division operation using class and if,elif..note:

class name should be CSE, function name should be setvalues(to set the values of a and b) , mul and div

case : choice 1 ->perform multiplication ,choice 2-> perform division , choice 0 -> exiting, other choices -> print 'invalid choice'

For example:

Input	Result
5 5 1 0	Result: 25 Exiting!

Answer: (penalty regime: 0 %)

```
class CSE:
    def setvalues(self, a, b):
        self.a = a
        self.b = b
    def mul(self):
        print("Result: ", self.a * self.b)
    def div(self):
        print("Result: ", self.a // self.b)
a= int(input())
b= int(input())
obj = CSE()
obj.setvalues(a,b)

while True:
    choice = int(input())
    if choice == 1:
        obj.mul()
    elif choice == 2:
        obj.div()
    elif choice == 0:
        print("Exiting!")
        break
```

	Input	Expected	Got	
✔	5 5 1 0	Result: 25 Exiting!	Result: 25 Exiting!	✔
✔	5 5 2 0	Result: 1 Exiting!	Result: 1 Exiting!	✔

Passed all tests! ✔

Correct

Marks for this submission: 20.00/20.00.

Question **5**

Incorrect

Mark 0.00 out of 20.00

Write a program in Python that asks the user to enter ten integers of their choice and return them a dictionary whose keys are the entered integers and whose values are 'prime' or 'not prime' depending on the entered integer.

For example:

Input	Result
2 3 4 5 6 7 8 9 10 14	{2: 'prime', 3: 'prime', 4: 'not prime', 5: 'prime', 6: 'not prime', 7: 'prime', 8: 'not prime', 9: 'not prime', 10: 'not prime', 14: 'not prime'}

Answer: (penalty regime: 0 %)

```
def is_prime(n):
    if n < 2:
        return False
    for i in range(2, int(n ** 0.5)+1):
        if n % 1 == 0:
            return False
    return True
prime_dict = {}
for _ in range(10):
    num = int(input())
    if is_prime(num):
        prime_dict[num]="prime"
    else:
        prime_dict[num]= "not prime"
print(prime_dict)
```

	Input	Expected	Got	
✖	2 3 4 5 6 7 8 9 10 14	{2: 'prime', 3: 'prime', 4: 'not prime', 5: 'prime', 6: 'not prime', 7: 'prime', 8: 'not prime', 9: 'not prime', 10: 'not prime', 14: 'not prime'}	{2: 'prime', 3: 'prime', 4: 'not prime', 5: 'not prime', 6: 'not prime', 7: 'not prime', 8: 'not prime', 9: 'not prime', 10: 'not prime', 14: 'not prime'}	✖

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.