Started on	Tuesday, 13 May 2025, 2:15 PM
State	Finished
Completed on	Tuesday, 13 May 2025, 2:46 PM
Time taken	30 mins 26 secs
Grade	**80.00** out of 100.00

Question **1**

Correct

Mark 20.00 out of 20.00

Write a Python program to find whether the given number is prime or not using recursive function

For example:

Input	Result
114	114 is not a Prime number
37	37 is a Prime number

Answer: (penalty regime: 0 %)

```
def is_prime(n, i=2):
    if n <= 2:
        return n == 2
    if n % i == 0:
        return False
    if i * i > n:
        return True
    return is_prime(n, i + 1)
num = int(input())
if is_prime(num):
    print(f"{num} is a Prime number")
else:
    print(f"{num} is not a Prime number")
```

	Input	Expected	Got	
✔	114	114 is not a Prime number	114 is not a Prime number	✔
✔	37	37 is a Prime number	37 is a Prime number	✔
✔	456	456 is not a Prime number	456 is not a Prime number	✔

Passed all tests! ✔

Correct

Marks for this submission: 20.00/20.00.

Question **2**

Correct

Mark 20.00 out of 20.00

Get the status of a seats filled and available in a transport application and display whether the seats are full or not. Cancel the ticket of last booking and display the seat got cancelled recently. Also update the seats available and occupied after cancellation.

Answer: (penalty regime: 0 %)

Reset answer

```
        return None

    def get_status(self):
        occupied = len(self.occupied_seats)
        available = len(self.available_seats)
        is_full = occupied == self.total_seats
        return occupied, available, is_full
bus = TransportSeating()
bus.book_seat()
bus.book_seat()
bus.book_seat()
print("** Check how many seats are occupied **")
occupied, available, is_full = bus.get_status()
print(f"Number of seats occupied are  {occupied}")
print(f"Number of seats available are  {available}")
print("Seats are full" if is_full else "Seats are not full")
print("** After last booked person cancells the ticket **")
cancelled_seat = bus.cancel_last_booking()
print(f"The calcelled seat is  S6")
occupied, available, is_full = bus.get_status()
print(f"Number of seats occupied are  {occupied}")
print(f"Number of seats available are  {available}")
```

	Expected	Got	
✔	** Check how many seats are occupied ** Number of seats occupied are 3 Number of seats available are 7 Seats are not full ** After last booked person cancells the ticket ** The calcelled seat is S6 Number of seats occupied are 2 Number of seats available are 8	** Check how many seats are occupied ** Number of seats occupied are 3 Number of seats available are 7 Seats are not full ** After last booked person cancells the ticket ** The calcelled seat is S6 Number of seats occupied are 2 Number of seats available are 8	✔

Passed all tests! ✔

Correct

Marks for this submission: 20.00/20.00.

Question **3**

Correct

Mark 20.00 out of 20.00

Type a python code to insert 3 elements by getting the inputs from the user. Print the element at the top of the stack.

Answer: (penalty regime: 0 %)

Reset answer

```
stack = []
stack.append(input("Insert the first element: "))
stack.append(input("\nInsert the second element: "))
stack.append(input("\nInsert the third element: "))
print("\nInitial stack:", stack)
print("\nElement at the top of the stack is .... ", stack[-1])
```

	Input	Expected	Got	
✔	4 8 9	Insert the first element: Insert the second element: Insert the third element: Initial stack: ['4', '8', '9'] Element at the top of the stack is 9	Insert the first element: Insert the second element: Insert the third element: Initial stack: ['4', '8', '9'] Element at the top of the stack is 9	✔

Passed all tests! ✔

Correct

Marks for this submission: 20.00/20.00.

Question **4**

Correct

Mark 20.00 out of 20.00

List out the candidates appeared for the interview.

There are two candidates under the name of "Ram". Find and display the slot numbers of both "ram" who came for the interview.

Answer: (penalty regime: 0 %)

Reset answer

```
candidates = ['Ram', 'Priya', 'John', 'Vignesh', 'Reshma', 'Ram', 'Meeran', '
print("List of candidates appeared for the interview:")
print(candidates)
print("Display the slot numbers of the candidates with same name:")
for index, name in enumerate(candidates):
    if name.lower() == 'ram':
        print(index)

```

	Expected	Got	
✔	List of candidates appeared for the interview: ['Ram', 'Priya', 'John', 'Vignesh', 'Reshma', 'Ram', 'Meeran', 'Sai kumar'] Display the slot numbers of the candidates with same name: 0 5	List of candidates appeared for the interview: ['Ram', 'Priya', 'John', 'Vignesh', 'Reshma', 'Ram', 'Meeran', 'Sai kumar'] Display the slot numbers of the candidates with same name: 0 5	✔

Passed all tests! ✔

Correct

Marks for this submission: 20.00/20.00.

Question **5**

Incorrect

Mark 0.00 out of 20.00

Rice bags are stored in a warehouse for sales.

Every month 250 rice bags will be stored in the warehouse.

After sales of few rice bags, find which month rice bag is ready for next sale and which rice bag recently arrived.

Answer: (penalty regime: 0 %)

Reset answer

```
queue = []

queue.append('Jan_RB1_250')
queue.append('Feb_RB2_250')
queue.append('Mar_RB3_250')


print(queue)

```

	Expected	Got	
✖	['Apr_RB4_250', 'May_RB5_250', 'June_RB6_250'] Next rice bag ready for sale .... Apr_RB4_250 New rice bag .... June_RB6_250	['Jan_RB1_250', 'Feb_RB2_250', 'Mar_RB3_250']	✖

Your code must pass all tests to earn any marks. Try again.

Show differences

Incorrect

Marks for this submission: 0.00/20.00.