

Question:1

```
def prime(num):  
    for i in range(2, num):  
        if num % i == 0:  
            return 0  
    else:  
        return num
```

while True:

```
n = int(input("Enter the number: "))
```

```
if n == 0:
```

```
    print("\nLoop ends here.... \n\n Here we are  
    printing all prime number from 1 to 100 using  
    prime function: \n\n")
```

```
    break
```

```
a = prime(n)
```

```
if a != 0:
```

```
    print(n, "is prime number \n")
```

```
else:
```

```
    print(n, "is not prime number \n")
```

```
def prime(n):
```

```
    status = True
```

```
    if n < 2:
```

```
        status = False
```

```
    else:
```

```
        for i in range(2, n):
```

```
            if n % i == 0:
```

```
                status = False
```

```
    return status
```

Problem Solving Using Python and R Lab

Lab3. Python Functions and Modules

Question 1. Create a function `prime()` that receives an integer and returns whether `n` is prime or not. Print all prime numbers from 1 to 100 by calling `prime()` function. For example,

```
prime(1)
prime(2)
.....
prime(100)
```

Q:1

```
for n in range(1, 100):
    if prime(n):
        if n == 100:
            print('prime', n)
        else:
            print('prime', n)
```

Question 2. Develop a simple arithmetic calculator for 4 operations. The program should continue calculation until user types 'q' to quit. A sample user interaction can be:

- Enter operator (q to quit): +
- Enter value 1: 10
- Enter value 2: 20
- Result = 30

Create 4 functions `add()`, `subtract()`, `multiply()` and `divide()` that receives two values and returns the result of the operation.

Now, perform the following operations by calling the corresponding functions. Validate your outputs.

1. 10+20
2. 20-5
3. 8*5
4. 50/3

Question: 2

```
def add(a, b):
    return a + b

def sub(a, b):
    return a - b

def mul(a, b):
    return a * b

def div(a, b):
    if b == 0:
        print("division is not possible")
    else:
        return a / b
```

Question: 2

While True:

```
print("Enter the operator: ")
```

```
operator = input()
```

```
if (operator == 'q'):
```

```
    break
```

```
a = int(input("\nEnter the value of 1: "))
```

```
b = int(input("\nEnter the value of 2: "))
```

```
if (operator == '+'):
```

```
    r = add(a, b)
```

```
elif (operator == '-')
```

```
    r = sub(a, b)
```

```
elif (operator == '*')
```

```
    mul  
    r = sub(a, b)
```

```
elif (operator == '/')
```

```
    r = div(a, b)
```

```
print("Result is ", r, '\n')
```

```
print("Quit")
```

Question3. Create a function `factorial()` that takes an integer and returns its factorial value.

- You can create as a non-recursive version of factorial.
- Also, check factorial of negative number does not exist.
- Factorial of 0 is 1.
- Save this Python file as `factorial_definition.py`.

Now, open another file and you can import `factorial_definition.py` as follows:

- `import factorial_definition`
- You can call factorial function as `factorial_definition.factorial()`.

Now, print the following factorial values:

1. `factorial_definition.factorial(3)`
2. `factorial_definition.factorial(5)`
3. `factorial_definition.factorial(10)`

Question:3

```
def factorial(n)
    result = 1;
    if (n < 0):
        print("The function does not exist for negative number")
    elif (n == 0):
        print("The factorial of 0 is 1")
    elif (n == 1):
        return 1
    else:
        for i in range(2, n+1):
            result = result * i;
        return result;

import factorial_definition
factorial_definition.factorial(3)
factorial_definition.factorial(5)
factorial_definition.factorial(10)
```