NAME: **Abid Hussain**   I.D#**12929**

Subject: Data Sciences

BS (CS,SE)   Instructor: M.Ayub Khan

Note:

At the top of the answer sheet there must be the ID, Name and semester of the concerned Student.

Students must have to provide the output of their respective programs. Students have same answers or programs will be considered fail. Programs in Python and codes should be explained clearly.

As this assignment is online so incase of any ambiguity my Whatsapp no. is 034499121116.

Q1. a. Why Functions are used discuss in detail?

 ANS: **Functions:** A function is a block of organized code that is used to perform a single related action. It provide better modularity for your application and a high degree of code. You have already seen various functions like printf() and main(). These are called built-in functions but we can write our own functions as well.

  b. How arguments are used in function , write a simple program in Python?

 ANS: It is the value that is sent to the function when it is called.
        printf("Abid study" , "in university")

        Output: Abid study in university.

Q2. a. Why upper(), lower(), capitalize() and swapcase() function are used ?

 ANS: The **upper**() function converts lowercase letters to uppercase letters.

```
>> upper("ABC abc 123")

    'ABC ABC 123'
```

The **lower()** function converts uppercase letters to lowercase letters.

```
>> lower("ABC abc 123")

    'abc abc 123'
```

The **capitalize()** function capitalizes a given word in a string.

```
>> capitalize("bill")

    'Bill'
```

The **swapcases()** function converts uppercase to lowercase letters vice versa.

```
>> swapcase("ABC abc 123")

    'abc ABC 123'
```

b. Write a program in which the discussed functions are used.
   Note : Q2 part a functions.

ANS: **Upper():**
        String = " today is mondy!"
        Print(string.upper())

Output: TODAY IS MONDAY!

Lower():
string= "TODAY IS MONDAY!"
print(string.lower())
Output: today is monday!

Capitalize():
string = " python is gReat"
capitalized-string=string.capitalize()
print('Old String: ` , string)
print(`Capitalized String: ` , capitalized_string)
Output: Old String: python is GREAT
Capitalized String: Python is great

swapcase():
string= "python is great."
print(string.swapcase())
Output: PYTHON IS GREAT.

Q3. a. What are the rules for defining the function?

ANS: Here are simple rules to define a function in Python.

1. Function blocks begin with the keyword followed by the function name and parentheses (( )).
2. Any input should be placed within these parameters or arguments parentheses.
3. The first statement of a function can be an optional statement.
4. The code block within every function starts with a colon (:).
5. The statement return, exits a function, optionally passing back an expression to the caller.
6. A return statement with no arguments is the same as return None.

 b. Write a suitable program of our defined function in Python?

ANS: def our_function():
        print("This is our new defined function")
        print("that's great")
    our_function()

Q4. a. What are the rules for defining the function and Parameter passing to the function?

ANS: When the arguments are passed by the value that is when the function is called the parameter receives a copy of the arguments in which address and this rule applies to all the scaler values structure and passed as arguments that does not modify the corresponding argument by the function call.

b. Write a suitable program of our defined function by parameter passing in Python?

ANS: def our_function(phrase):
    print(phrase)
    own_function("today is rainning")

Output: today is raining

Q5. a. What are return values to a Function discuss in detail?

ANS: It is the statement which is used to the end the execution of the function call and returns the result to the caller the statement after the return statements are not executed and if the return statement is without any expression.
Syntax: def fun(): statements . . return [expression]

b. Write a suitable program of a Function with returning value?

ANS: // C code for function with no arguments

    // but have return value
    #include <math.h>
    #include <stdio.h>

```c
    int sum();
    int main()
    {
        int num;
        num = sum();
    printf("\nSum of two given values = %d", num);
    return 0;
}

int sum()
{
    int a = 50, b = 80, sum;
    sum = sqrt(a) + sqrt(b);
    return sum;
}
```

Output: Sum of two given values=16