### **FUNCTIONS EXERCISE**

### **Exercise 1: Create a function in Python**

Write a program to create a function that takes two arguments, name and age, and print their value.

ANS: def my-function(name,age)

Print(name+” “ +age)

My\_function(“Bharath”,”18)

### **Exercise 2: Create a function with variable length of arguments**

Write a program to create function func1() to accept a variable length of arguments and print their value.

**Note**: Create a function in such a way that we can pass any number of arguments to this function, and the function should process them and display each argument’s value.

**Function call**:

# call function with 3 arguments

func1(20, 40, 60)

# call function with 2 arguments

func1(80, 100)

**Expected Output**:

Printing values

20

40

60

Printing values

80

100

ANS: def func1(\*args):

For i in args:

Print(i)

Func1(20,40,60)

Func2(80,100)

### **Exercise 3: Return multiple values from a function**

Write a program to create function calculation() such that it can accept two variables and calculate addition and subtraction. Also, it must **return both addition and subtraction in a single return call**.

**Given**:

def calculation(a, b):

# Your Code

res = calculation(40, 10)

print(res)

**Expected Output**

50, 30

**Expected Output**:

ANS:

|  |
| --- |
|  |
| def calculation(a, b): |
|  | addition = a + b |
|  | subtraction = a - b |
|  | return addition, subtraction |
|  | res = calculation(40, 10) |
|  | print(res) |
|  |  |

### **Exercise 4: Create a function with a default argument**

Write a program to create a function show\_employee() using the following conditions.

* It should accept the employee’s name and salary and display both.
* If the salary is missing in the function call then assign default value 9000 to salary

**Given**:

showEmployee("Ben", 12000)

showEmployee("Jessa")

**Expected output**:

Name: Ben salary: 12000

Name: Jessa salary: 9000

ANS:def showEmpolyee(name,salary=9000):

Print(name+” “+salary)

Show\_empolyee(“ben”,”12000”)

Show\_empolyee(“jessa”)

### **Exercise 5: Create an inner function to calculate the addition in the following way**

* Create an outer function that will accept two parameters, a and b
* Create an inner function inside an outer function that will calculate the addition of a and b
* At last, an outer function will add 5 into addition and return it

ANS: def outer\_fun(a, b):

square = a \*\* 2

def addition(a, b):

return a + b

add = addition(a, b)

return add + 5

result = outer\_fun(5, 10)

print(result)

### **Exercise 6: Create a recursive function**

Write a program to create a **recursive function to calculate the sum of numbers** from 0 to 10.

A recursive function is a function that calls itself again and again.

**Expected Output**:

55

Ans: def addition(num):

if num:

return num + addition(num - 1)

else:

return 0

res = addition(10)

print(res)

### **Exercise 7: Assign a different name to function and call it through the new name**

Below is the function display\_student(name, age). Assign a new name show\_tudent(name, age) to it and call it using the new name.

**Given**:

def display\_student(name, age):

print(name, age)

display\_student("Emma", 26)

You should be able to call the same function using

show\_student(name, age)

Ans:

|  |
| --- |
| def display\_student(name, age): |
|  | print(name, age) |
|  |  |
|  | display\_student("Emma", 26) |
|  |  |
|  | showStudent = display\_student |
|  | showStudent("Emma", 26) |
|  |  |

### **Exercise 8: Generate a Python list of all the even numbers between 4 to 30**

**Expected Output**:

[4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28]

Print(list(range(4,30,2)))

### **Exercise 9: Find the largest item from a given list**

x = [4, 6, 8, 24, 12, 2]

**Expected Output**:

24

Ans: print(max(x))