**10211CS213- Python Programming**

**Task 3 : Importing Python modules and packages in python programming**

**3.1 Math module**

Ram given following maths calculation task to Dhina.

To find

1. base 10 logarithm of the number 13
2. power of the 10 corresponding to the value of 2
3. The floor value of the 10.25201
4. The ceil value of the 10.25201
5. The absolute value of -10.024
6. The factorial of the given number 5.

Constraint: Use math module.

# **Aim**

Write a program to perform mathematical calculation using math module

**Algorithm:**

Step 1 : Open python script file.

Step 2 : import math module.

Step 3: math.log10 used to find log base 10

Step 4: \*\* used to calculate power

Step 5: math.floor(),math.ceil() used to calculate floor and ceil value

step 6: abs() method used to display absolute value

step 7: factorial() method to display the factorial value

**Source code:**

import math

print("Logarithm",math.log10(13))

print("Power",math.pow(10,2))

print("Floor",math.floor(10.25201))

print("Ceil",math.ceil(10.25201))

print("absolute", abs(-10.024))

print("Factorial",math.factorial(5))

**Output:**

Logarithm 1.1139433523068367

Power 100.0

Floor 10

Ceil 11

absolute 10.024

Factorial 120

**3.2 Total weight calculation**

An online retailer sells two products: widgets and gizmos. Each widget weighs 75 grams. Each gizmo weighs 112 grams. Write a program that reads the number of widgets and the number of gizmos from the user. Then your program should compute and display the total weight of the parts using the concept of packages and modules.

**Aim**

To write a Python program using **packages and modules** that reads the number of widgets and gizmos sold by an online retailer, calculates their **total weight** based on predefined weights, and displays the result to the user.

**Algorithm**

1. **Start**
2. Create a **package folder** named shop with:
   * A module file weight\_calc.py containing a function to calculate total weight.
   * An \_\_init\_\_.py file to make it a package.
3. In weight\_calc.py,
   * Define constants: WIDGET\_WEIGHT = 75 grams and GIZMO\_WEIGHT = 112 grams.
   * Create a function calculate\_total\_weight(widgets, gizmos) that:
     + Multiplies the number of widgets by WIDGET\_WEIGHT.
     + Multiplies the number of gizmos by GIZMO\_WEIGHT.
     + Adds the two results to get total\_weight.
     + Returns total\_weight.
4. In the main program (main.py):
   * Import the weight\_calc module from the shop package.
   * Prompt the user to **enter the number of widgets**.
   * Prompt the user to **enter the number of gizmos**.
   * Call the calculate\_total\_weight() function with these inputs.
   * Display the total weight in grams.
5. **End**

Program

Here’s the structure:

shop/

\_\_init\_\_.py

weight\_calc.py

main.py

**1. shop/weight\_calc.py**

# weight\_calc.py

def calculate\_total\_weight(widgets, gizmos):

WIDGET\_WEIGHT = 75 # grams

GIZMO\_WEIGHT = 112 # grams

total\_weight = (widgets \* WIDGET\_WEIGHT) + (gizmos \* GIZMO\_WEIGHT)

return total\_weight

**2. shop/init.py**

# \_\_init\_\_.py

# This makes 'shop' a package

**3. main.py**

# main.py

from shop import weight\_calc

def main():

widgets = int(input("Enter the number of widgets: "))

gizmos = int(input("Enter the number of gizmos: "))

total\_weight = weight\_calc.calculate\_total\_weight(widgets, gizmos)

print(f"The total weight of the parts is {total\_weight} grams.")

if \_\_name\_\_ == "\_\_main\_\_":

main()

**Sample Input and Output**

Enter the number of widgets: 5

Enter the number of gizmos: 3

The total weight of the parts is 731 grams.

**3.3 OS and Sys modules**

Akhil bought a new system that time his class mates ask operating system details like name, getcwd, listdir and getlogin and system details like system, release, platform details. Help Akhil to find all details using OS and sys modules.

# Aim

Write a program to display os and sys details using os and sys module

**Algorithm:**

Step 1 : Open python script file.

Step 2 : import os and sys module.

Step 3: os.name to display the name of the os

Step 4: os.getcwd() to display current working directory

Step 5: os.listdir() to display the directory list details

step 6: sys.platform used to display linux platform

step 7: platform.system(),platform.release())method to display the platform details and platform release details

**Source code:**

import os

import sys

import platform

import sysconfig

print("os name", os.name)

print(os.getcwd())

print(os.listdir())

print(os.getlogin())

print("platform name", sys.platform)

print("platform.system()", platform.system())

print("platform.release()", platform.release())

**Output:**

os.name nt

/tmp/sessions/8ab062fead14b3db

['main.py']

<built-in function getlogin>

posix.uname\_result(sysname='Linux', nodename='5c10b58a99fb', release='5.11.0-1017-gcp', version='#19~20.04.1-Ubuntu SMP Thu Aug 12 05:25:25 UTC 2021', machine='x86\_64')

sys.platform Win32

platform.system() Windows

sys.release() 11

**3.4 Random wish**

Illustrate a program Shiva wish to Srini. The program should display unique number wishes, at random, each time its run using random module

# **Aim**

Write a program to display unique number wishes

**Algorithm:**

Step 1 : Open python script file.

Step 2 : import random module

Step 3: list the wishes in fortunes

Step 4: random.choice() using to display the random wishes

**Source code:**

import random

fortunes = ["Good things come to those who wait.",

"Patience is a virtue.",

"The early bird gets the worm.",

"A wise man once said, everything in its own time and place.",

"Fortune cookies rarely share fortunes."]

print (random.choice(fortunes))

**Sample Output 1**

Patience is a virtue.

**Sample Output 2**

The early bird gets the worm.

3.5 **Mathematical and String operations**

**Aim**

To write a Python program using **packages** and **modules** that performs mathematical operations (addition, multiplication, factorial) and string operations (reverse a string, count vowels) by organizing code into reusable components.

**Algorithm**

**Step 1 – Create the Package Structure**

1. Create a folder named my\_package.
2. Inside my\_package, create the files:
   * \_\_init\_\_.py
   * math\_utils.py
   * string\_utils.py

**Step 2 – Implement math\_utils.py Module**

1. Define a function add(a, b) to return the sum of two numbers.
2. Define a function multiply(a, b) to return the product of two numbers.
3. Define a function factorial(n) that:
   * Returns 1 if n is 0 or 1.
   * Otherwise, returns n \* factorial(n-1) (recursive call).

**Step 3 – Implement string\_utils.py Module**

1. Define a function reverse\_string(s) that returns the reversed string using

slicing (s[::-1]).

1. Define a function count\_vowels(s) that:
   * Stores vowels in a string "aeiouAEIOU".
   * Counts characters from the input string that are present in vowels.

**Step 4 – Initialize Package with \_\_init\_\_.py**

1. Import functions from math\_utils and string\_utils into \_\_init\_\_.py for easy access.

**Step 5 – Create main.py Program**

1. Import required functions from my\_package.
2. Call and display results of:
   * Addition, multiplication, factorial from math module.
   * Reverse string and vowel counting from string module.

**Step 6 – Execute the Program**

1. Run python main.py in the terminal.
2. Verify that outputs match expected results.

**Program**

1. Folder structure:

my\_package/

\_\_init\_\_.py

math\_utils.py

string\_utils.py

main.py

my\_package/math\_utils.py

# math\_utils.py

def add(a, b):

return a + b

def multiply(a, b):

return a \* b

def factorial(n):

if n == 0 or n == 1:

return 1

return n \* factorial(n - 1)

my\_package/string\_utils.py

# string\_utils.py

def reverse\_string(s):

return s[::-1]

def count\_vowels(s):

vowels = 'aeiouAEIOU'

return sum(1 for char in s if char in vowels)

my\_package/\_\_init\_\_.py

# \_\_init\_\_.py

from .math\_utils import add, multiply, factorial

from .string\_utils import reverse\_string, count\_vowels

(This makes it easy to access all functions directly from my\_package.)

2. Main Program (main.py)

# main.py

from my\_package import add, multiply, factorial, reverse\_string, count\_vowels

# Math functions

print("Addition:", add(10, 5))

print("Multiplication:", multiply(3, 4))

print("Factorial of 5:", factorial(5))

# String functions

print("Reversed string:", reverse\_string("Python"))

print("Number of vowels:", count\_vowels("Hello World"))

**Sample Output:**

Addition: 15

Multiplication: 12

Factorial of 5: 120

Reversed string: nohtyP

Number of vowels: 3