**PYTHON CODES**

**Task 1: Basic Data Types**

# Task 1: Basic Data Types

x = 5 # Integer

y = 2.5 # Float

# Perform operations

add\_result = x + y

sub\_result = y - x

mul\_result = x \* y

power\_result = x \*\* 2

floor\_div\_result = x // 2

# Print the results

print("Addition of x and y:", add\_result)

print("Subtraction of y from x:", sub\_result)

print("Multiplication of x and y:", mul\_result)

print("x raised to the power of 2:", power\_result)

print("Floor division of x by 2:", floor\_div\_result)

**Output: -**

Addition of x and y: 7.5

Subtraction of y from x: -2.5

Multiplication of x and y: 12.5

x raised to the power of 2: 25

Floor division of x by 2: 2

**Task 2: Lists**

# Task 2: Lists

my\_list = [1, 2, 3, 4, 5]

# Replace the third element with "hello"

my\_list[2] = "hello"

# Add "world" to the end of the list

my\_list.append("world")

# Remove the first element from the list

my\_list.pop(0)

# Print the final list

print("Final list:", my\_list)

**Output: -**

Final list: [2, 'hello', 4, 5, 'world']

**Task 3: Dictionaries**

# Task 3: Dictionaries

student\_scores = {'Alice': 85, 'Bob': 90, 'Charlie': 78}

# Add 'David' with a score of 88

student\_scores['David'] = 88

# Update 'Alice''s score to 95

student\_scores['Alice'] = 95

# Delete 'Charlie' from the dictionary

del student\_scores['Charlie']

# Print the final dictionary

print("Final student scores:", student\_scores)

**Output: -**

Final student scores: {'Alice': 95, 'Bob': 90, 'David': 88}

**Task 4: Functions**

# Task 4: Functions

def calculate\_area(width, height):

return width \* height

# Call the function with width = 5 and height = 10

area = calculate\_area(5, 10)

print("Area of the rectangle:", area)

**Output: -**

Area of the rectangle: 50

**Task 5: Classes and Inheritance**

# Task 5: Classes and Inheritance

class Animal:

def \_\_init\_\_(self, name):

self.name = name

def speak(self):

print("The animal speaks")

# Dog class inherits from Animal

class Dog(Animal):

def speak(self):

print("Woof! Woof!")

# Create an instance of Dog and call its speak method

buddy = Dog("Buddy")

buddy.speak()

**Output: -**

Woof! Woof!