OBJECTIVES

1) What is the primary purpose of classes and objecttives in python:, A

A) The primary purpose of classes and objects in Python is to organise Code into reusable code

2) Which Keyword is used to implement inheritance in Python.

B: Inherit

3) Which of the following statements about a Variable scope in Python is true.

A and D :

The Answere to question 3 would be A and D.

4) What is the Purpose of formatting strings in Python.

C : To replace placeholders with Variable Values

5) What is the main purpose of using the try-except in python.

C : They handle errors and Exceptions gracefully.

Success has a simple formular do your best, and people may like it.

Code Questions

Write a python class named 'Rectangle' with attributes 'length' and width and a method area () and returns the area of the rectangle.

Python Code:

```
Class Rectangle:
# This class represents a rectangle with attributes
  length and width
def __init__(self, length, width):
    # Initialize a rectangle object with the given length
and width
    self. length = length
    self. Width = width
def area(self)
    # Calculate and returns the area of the rectangle
    # Returns the area of the rectangle (long * width)
return self. length * self. Width

# Example usage
my-rectangle = Rectangle (5,3)
area = my-rectangle. area ()

print (f"The area of the rectangle is : {area}")
```

7) Define a class 'Square' that inherits from the Rectangle class defined in question [illegible]. Add a method perimeter() to Calculate the perimeter of the square.

Python Code

```
~~Class~~
    Rectangle:
  # This class represents a rectangle with attributes
    length and width.
  def __init(self, length, width):
    # initializes a given rectangle object with the
      given length and width
   Self.length = length
   Self.width = width
  def  area (self):
    # Calculates and returns the area.
    # Returns the area of the rectangle (L*W)
    return self.length * self.width

class  square(Rectangle)
  # This class represents a square which is a
    special type of rectangle with equal sides.

 def __init__(self, side_length)
   # Initializes a square object with a
     given side length.
    # side_length -> The length of side of
      square
 super().~~init~~__init__(side-length, side-length)
# Calls the parent Constructor with same
  side length for both length and width
```

```
def perimeter (self):
    # Calculates and returns the perimeter of the
    square (4 * side_length)
return 4 * self.length # since length and width are
the same for square.

Getting user input for the square.
While True:
    try:
    side_length = float(input("Enter the side length of
                         the square: "))
      if side_length > 0:
          break
      else:
    print("The side length must be a positive number.")

    except ValueError:
    print("Invalid input: Please enter a number only")
  # Create a square object.
  my_square = Square(side_length)

 # Calculate and print the square's area and the perimeter.
  area = my_square.area()
  perimeter = my_square.perimeter()
print(f"The area of the square is: {area}")

print(f"The perimeter of the square is: {perimeter}")
```

8 Write a Python function 'calaulateAge()' that takes the year of birth as an input and returns the age. Handle any potential errors using the try except block

```
def calculateAge(birth_year):
 # Calculates and returns the age based on the
 given birth year.
 # Returns the current age of the person [Current_year - Birth_year]
 # Raises:
   ValueError:-> If the entered value is not a
   valid interger.
 try:
 # Get current year
 current_year = date.today().year.
 age =       current_year - birth_year
   if age < 0:
   raise ValueError("Invalid birth year")
   return age
   except ValueError as e:
   print(f"Error: {e}")
    return None
# Get the user input
 While True:
   try:
   birth_year = int(input("Enter year of birth:"))
    break
   except ValueError:
   print("Invalid input. Please Enter an Interger
          year")
```

```
# Calculate and display age :
age =  CalculateAge ( birth_year)

If age is not None :
print (f" Your age is : {age} ")
```

⑨ Write a Python Program that prompts the user to Enter their name and age then prints a greeting message along with their age.

```
def   get_positive_interger ( prompt )
# Prompts the user for a positive interger and Validate the input.
# Returns a positive interger returned by the user

While True:
    try:
  Value = int (input( prompt))
    If Value > 0:
    return Value.
    else:
    print ("Please Enter a positive interger.")
    except ValueError:
    print ("Invalid input. please enter a number only.")
# Getting user input.
  name = input ("Enter your name:")

# Get user age (positive interger)
    age  = get_positive_interger ("Enter your age")

print (f" Hello {name}, nice to meet someone who
      is {age} years old! "
```

(10) Define a Function 'divideNumber()', that takes two numbers as user input and returns the result of dividing the first number by the second number. Handle the ZeroDivisionError using the try-except block.

```
def divideNumber(num1, num2):
    # This function divides two numbers and
handles the ZeroDivisionError.
    Returns
  • The result of dividing num1 by num2, or a message if
  there is a ZeroDivisionError.
    try:
    return num1 / num2.
    except ZeroDivisionError:
      Return "Sorry, you can't divide by zero."
# Example usage.
  While True:
    try:
  num1 = float(input("Enter the first number: "))
  num2 = float(input("Enter the second number: "))
  break
  except ValueError:
 print("please enter valid numbers")

result = divideNumber(num1, num2)

print(result)
```