Write a menu-driven program to perform swap case and reverse operations of the string given by the user using the concept of classes.

The functionalities of the program:

swap case: Swaps cases, the lower case becomes the upper case and vice versa. reverse: return the reversed version of the string. You need to display the string after performing the selected operation.

Constraints:

The input string should only contain the alphabets.

Note:

Understand the Menu-driven format which is displayed in the test cases. The class name is given in the non-editable code itself.

```
class StringOperations:
    def __init__(self, text):
        self.text = text

    def swap_case(self):
        # Swap the case of each character in the string
        return self.text.swapcase()

    def reverse(self):
        # Reverse the string
        return self.text[::-1]

# Main Program
while True:
    # Get input from user and validate it only contains alphabets
    text = input("Enter a string (alphabets only): ")
    if not text.isalpha():
        print("Invalid input. Please enter a string containing only alphabets.")
        continue

    # Create an instance of StringOperations with the given text
    string_op = StringOperations(text)

    # Display the menu
    print("\nMenu:")
    print("1. Swap Case")
    print("2. Reverse")
    print("3. Exit")

    # Get user's choice
    choice = input("Enter your choice (1/2/3): ")

    # Perform operations based on user's choice
    if choice == '1':
        print("Result after swap case:", string_op.swap_case())
    elif choice == '2':
        print("Result after reverse:", string_op.reverse())
    elif choice == '3':
        print("Exiting the program.")
        break
    else:
        print("Invalid choice. Please select 1, 2, or 3.")
```

```
Enter a string (alphabets only): bhupinder
```

```
Menu:
1. Swap Case
2. Reverse
3. Exit
Enter your choice (1/2/3): 2
Result after reverse: rednipuhb
Enter a string (alphabets only): singh

Menu:
1. Swap Case
2. Reverse
3. Exit
Enter your choice (1/2/3): 3
Exiting the program.
```

Create a class named Test that contains a method called power(). This method should take two parameters: the base number and the exponent, and it should print the result of raising the base to the power of the exponent.

Constraints:

1 <= numbers <= 100

Sample Test Case:

Input:

2 -----> Enter base

4 ------> Enter power

Output:

16 ------> Print the result

```
class Test:
    def power(self, base, exponent):
        # Calculate base raised to the power of exponent
        result = base ** exponent
        # Print the result
        print(result)

# Main program to take user input and use the Test class

# Taking inputs for base and exponent
base = int(input("Enter base (1-100): "))
exponent = int(input("Enter power (1-100): "))

# Check if inputs are within the specified range
if 1 <= base <= 100 and 1 <= exponent <= 100:
    # Create an instance of Test and call the power method
    test = Test()
    test.power(base, exponent)
else:
    print("Both base and power should be between 1 and 100.")
```

```
Enter base (1-100): 2
Enter power (1-100): 4
16
```

Your task is to:

Define a class shapes that use the init method to initialize the radius(r) of the circle, semi-circle, and sphere respectively. Also define three methods area_cir(), area_semi(), and area_sph(), that take the values of r from the init method and calculate the respective area. Create an object of class shapes to call the methods defined in class shapes.

Note:

Understand the menu-driven format as displayed in the test cases Print the result up to 2 decimal places Area of circle= πr2 , Area of semi-circle = πr2/2 , Area of sphere = 4πr2

Constraints:

radius r must be positive integers > 0.

The value of π is 3.14

Sample Test case:

1. circle ------> Display the menu to select the shape.
2. semi-circle
3. sphere

1 ------> Select the option

2 -------> Represents radius

12.56 -----> Area of the specified shape.

```
class Shapes:
    def __init__(self, r):
        if r <= 0:
            raise ValueError("Radius must be positive")
        self.radius = r

    def area_cir(self):
        pi = 3.14
        area = pi * self.radius**2
        return round(area, 2)

    def area_semi(self):
        pi = 3.14
        area = pi * self.radius**2 / 2
        return round(area, 2)

    def area_sph(self):
        pi = 3.14
        area = 4 * pi * self.radius**2
        return round(area, 2)


if __name__ == "__main__":
    while True:
        print("1. Area of Circle")
        print("2. Area of Semi-Circle")
        print("3. Area of Sphere")
        print("4. Exit")
        choice = int(input("Enter your choice: "))

        if choice == 1:
            radius = int(input("Enter the radius of the circle: "))
            shape = Shapes(radius)
            print("Area of the circle:", shape.area_cir())
        elif choice == 2:
            radius = int(input("Enter the radius of the semi-circle: "))
            shape = Shapes(radius)
            print("Area of the semi-circle:", shape.area_semi())
        elif choice == 3:
            radius = int(input("Enter the radius of the sphere: "))
            shape = Shapes(radius)
            print("Area of the sphere:", shape.area_sph())
        elif choice == 4:
            break
        else:
            print("Invalid choice. Please try again.")
```

```
1. Area of Circle
2. Area of Semi-Circle
3. Area of Sphere
4. Exit
Enter your choice: 1
Enter the radius of the circle: 2
Area of the circle: 12.56
1. Area of Circle
2. Area of Semi-Circle
3. Area of Sphere
4. Exit
Enter your choice: 4
```

Create a class Test with a single method multiplication() that prints the multiplied value of two numbers.

Constraints:

1 <= numbers <= 100

Sample Test Case:

Input:

2 -----> Enter number 1.

6 ------> Enter number 2.

Output:

12 ------> Print the result

```
class Test:
    def multiplication(self, num1, num2):
        if 1 <= num1 <= 100 and 1 <= num2 <= 100:
            result = num1 * num2
            print(result)
        else:
            print("Numbers must be between 1 and 100")

# Create an object of the Test class
test_obj = Test()

# Get input from the user
num1 = int(input("Enter number 1: "))
num2 = int(input("Enter number 2: "))

# Call the multiplication method
test_obj.multiplication(num1, num2)
```

```
Enter number 1: 2
Enter number 2: 6
12
```

Your task is to:

Define a class Person, that uses init method to initialize two attributes: firstname and lastname of a person. Create an object of a class person and print only the last name of the person using the object.

Constraints:

1 <= length of the string <= 10

Note:

Refer to the displayed test cases for better understanding.

Sample Test Case:

Lincoln

Issac

Issac

```
class Person:
    def __init__(self, firstname, lastname):
        self.firstname = firstname
        self.lastname = lastname

# Create an object of the Person class
person1 = Person(input("Enter first name: "), input("Enter last name: "))

# Print the last name
print(person1.lastname)
```

```
Enter first name: Lincoln
Enter last name: Issac
Issac
```

Define a class Shapes that uses the **init**() method to initialize the length (l) and width (w) for different shapes: square, triangle, and cube.

Additionally, implement the following methods to calculate the respective areas:

area_sq(): for calculating the area of a square. area_tri(): for calculating the area of a triangle. area_cube(): for calculating the surface area of a cube.

Requirements:

Area of square: l^2 Area of triangle: \frac{1}{2} \times l \times w Surface area of cube: 6 \times l^2 Create an object of the Shapes class to call the methods and display the correct result based on the user's selection.

Note: If an invalid option is selected, the program should display "Invalid".

Constraints:

length l, width w must be positive integers > 0.

Sample Test Case:

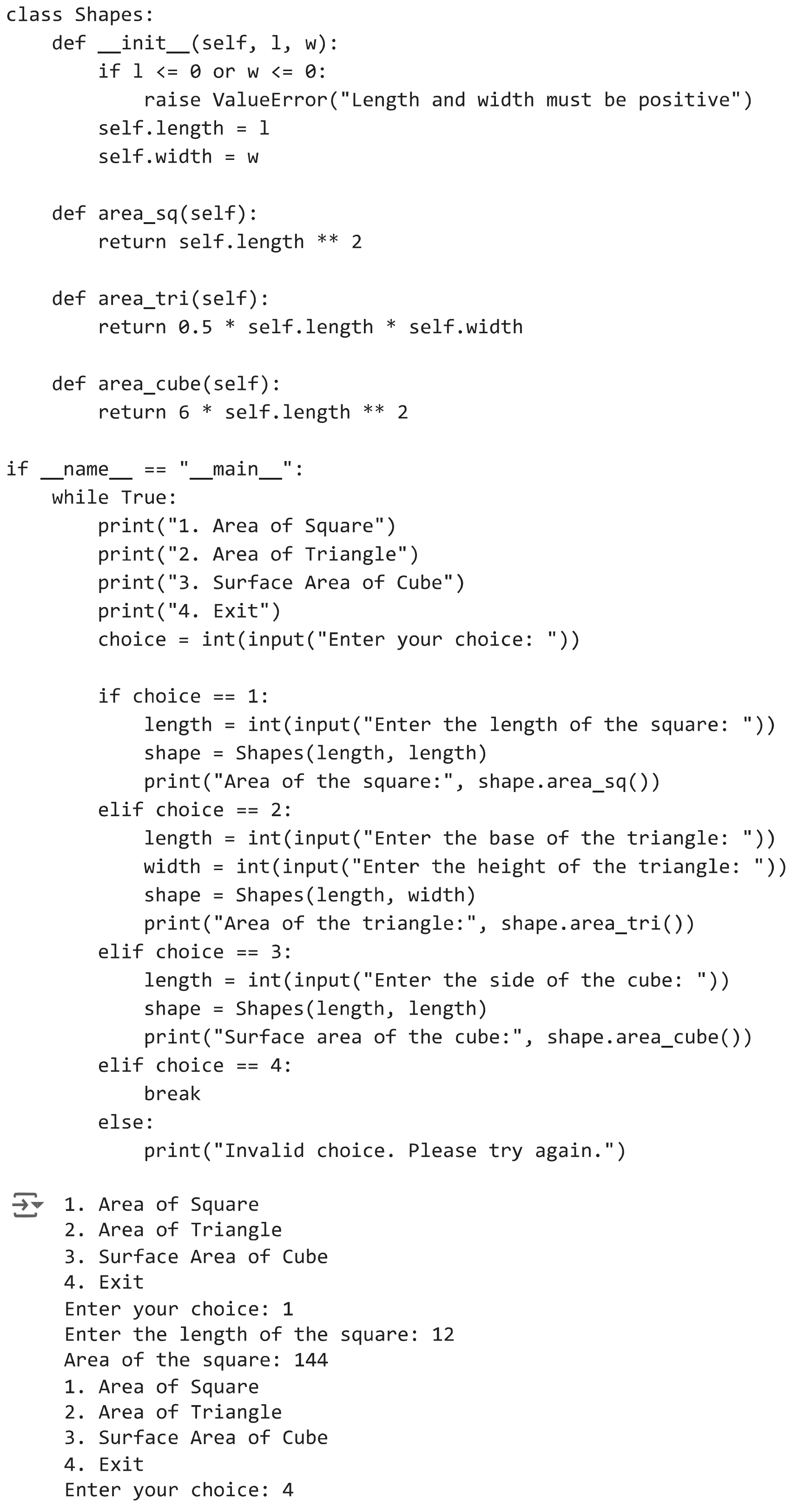
1. square ------> Display the menu to select the shape.
2. triangle
3. cube

2 ------> Select the option

10 -----> Represent's l

5 -----> Represent's w

25.0 ------> Area of the shape selected



```
class Shapes:
    def __init__(self, l, w):
        if l <= 0 or w <= 0:
            raise ValueError("Length and width must be positive")
        self.length = l
        self.width = w

    def area_sq(self):
        return self.length ** 2

    def area_tri(self):
        return 0.5 * self.length * self.width

    def area_cube(self):
        return 6 * self.length ** 2

if __name__ == "__main__":
    while True:
        print("1. Area of Square")
        print("2. Area of Triangle")
        print("3. Surface Area of Cube")
        print("4. Exit")
        choice = int(input("Enter your choice: "))

        if choice == 1:
            length = int(input("Enter the length of the square: "))
            shape = Shapes(length, length)
            print("Area of the square:", shape.area_sq())
        elif choice == 2:
            length = int(input("Enter the base of the triangle: "))
            width = int(input("Enter the height of the triangle: "))
            shape = Shapes(length, width)
            print("Area of the triangle:", shape.area_tri())
        elif choice == 3:
            length = int(input("Enter the side of the cube: "))
            shape = Shapes(length, length)
            print("Surface area of the cube:", shape.area_cube())
        elif choice == 4:
            break
        else:
            print("Invalid choice. Please try again.")
```

```
1. Area of Square
2. Area of Triangle
3. Surface Area of Cube
4. Exit
Enter your choice: 1
Enter the length of the square: 12
Area of the square: 144
1. Area of Square
2. Area of Triangle
3. Surface Area of Cube
4. Exit
Enter your choice: 4
```

Create a class Test with a single method division() that returns the division of two numbers.

Constraints:

1 <= numbers <= 100

Sample Test Case:

100 -----> Enter number 1.

2 ------> Enter number 2.

50 ------> Print the result

```
class Test:
    def division(self, num1, num2):
        if 1 <= num1 <= 100 and 1 <= num2 <= 100:
            result = num1 / num2
            return result
        else:
            print("Numbers must be between 1 and 100")

# Create an object of the Test class
test_obj = Test()

# Get input from the user
num1 = int(input("Enter number 1: "))
num2 = int(input("Enter number 2: "))

# Call the division method
result = test_obj.division(num1, num2)

if result is not None:
    print(result)
```

```
Enter number 1: 100
Enter number 2: 2
50.0
```