Assignment-2

Name   :   Balakumar Jayababu

College   :   Sri Venkateswara Engineering College.

Roll No/Hall Ticket :   209E1A3307

1.Write a Python program to calculate the area of a rectangle given its length and width

```python
def calculate_rectangle_area(length, width):
    area = length * width
    return area


# Taking user input for length and width
length = float(input("Enter the length of the rectangle: "))
width = float(input("Enter the width of the rectangle: "))


# Calculating the area
area = calculate_rectangle_area(length, width)


print("The area of the rectangle is:", area)
```

2.Write a program to convert miles to kilometers

```python
def miles_to_kilometers(miles):
    kilometers = miles * 1.60934
    return kilometers


# Taking user input for miles
miles = float(input("Enter the distance in miles: "))
```

3. Write a function to check if a given string is a palindrome

```python
def is_palindrome(s):
    s = s.lower()
    left = 0
    right = len(s) - 1
    while left < right:
        if not s[left].isalnum():
            left += 1
            continue
        if not s[right].isalnum():
            right -= 1
            continue
        if s[left] != s[right]:
            return False
        left += 1
        right -= 1
    return True


input_string = input("Enter a string: ")
if is_palindrome(input_string):
    print("Yes, it's a palindrome!")
else:
    print("No, it's not a palindrome.")
```

4. Write a Python program to find the second largest element in a list

```python
def second_largest(arr):
    if len(arr) < 2:
        return "List has less than two elements"

    largest = float('-inf')
    second_largest = float('-inf')

    for num in arr:
        if num > largest:
            second_largest = largest
            largest = num
        elif num > second_largest and num != largest:
            second_largest = num

    if second_largest == float('-inf'):
        return "There is no second largest element"
    else:
        return second_largest

# Test the function
my_list = [int(x) for x in input("Enter elements of the list separated by space: ").split()]
result = second_largest(my_list)
print("The second largest element in the list is:", result)
```

5. Explain what indentation means in Python.

In Python, indentation is used to define the structure and scope of code blocks. It's not just for visual clarity; it's a fundamental aspect of Python syntax. Here's what indentation signifies:

1. **Block Structure**: Indentation is used to denote blocks of code, such as those within loops, conditional statements, function definitions, and class definitions. Blocks of code at the same level of indentation are considered to be part of the same block or scope.

2. **Hierarchy**: Python uses indentation to establish the hierarchy of code. Code blocks that are indented to the same level are considered to be at the same hierarchical level. Blocks that are indented further are considered to be nested within the outer block.

3. **Readability**: Indentation enhances code readability by visually indicating the structure of the code. It helps programmers understand the flow and organization of the program more easily.

4. **Enforcement of Structure**: Unlike many other programming languages that use braces or keywords to denote blocks, Python relies solely on indentation to define block structure. This means that correct indentation is crucial for Python code to be syntactically valid.

6. Write a program to perform set difference operation

```python
def set_difference(set1, set2):
    return set1 - set2


# Example sets
set_a = {1, 2, 3, 4, 5}
set_b = {3, 4, 5, 6, 7}


# Perform set difference operation
result = set_difference(set_a, set_b)


print("Result of set difference:", result)
```

8. Write a program to calculate the factorial of a number using a while loop.

```python
def factorial(n):
    if n < 0:
        return "Factorial is not defined for negative numbers"
    elif n == 0:
        return 1
    else:
        result = 1
        while n > 0:
            result *= n
            n -= 1
        return result


# Test the function
number = int(input("Enter a number to calculate its factorial: "))
fact = factorial(number)
print("Factorial of", number, "is", fact)
```

9. Write a Python program to check if a number is positive, negative, or zero using if-elif-else statements.

```python
def check_number(num):
    if num > 0:
        return "Positive"
    elif num < 0:
        return "Negative"
    else:
        return "Zero"


# Test the function
number = float(input("Enter a number: "))
result = check_number(number)
print("The number is", result)
```

10. Write a program to determine the largest among three numbers using conditional statements

```python
def find_largest(num1, num2, num3):
    if num1 >= num2 and num1 >= num3:
        return num1
    elif num2 >= num1 and num2 >= num3:
        return num2
    else:
        return num3


# Test the function
number1 = float(input("Enter the first number: "))
number2 = float(input("Enter the second number: "))
number3 = float(input("Enter the third number: "))


largest = find_largest(number1, number2, number3)
print("The largest number is:", largest)
```

11. Write a Python program to create a numpy array filled with ones of given shape.

```python
import numpy as np


def create_ones_array(shape):
    ones_array = np.ones(shape)
    return ones_array


# Test the function
shape = tuple(map(int, input("Enter the shape of the array (comma-separated): ").split(',')))
ones_array = create_ones_array(shape)
print("Array filled with ones of shape", shape, ":\n", ones_array)
```

12. Write a program to create a 2D numpy array initialized with random integers.

```python
import numpy as np

def create_random_int_array(rows, cols, min_val, max_val):
    random_int_array = np.random.randint(min_val, max_val+1, size=(rows, cols))
    return random_int_array

# Test the function
rows = int(input("Enter the number of rows: "))
cols = int(input("Enter the number of columns: "))
min_val = int(input("Enter the minimum value for random integers: "))
max_val = int(input("Enter the maximum value for random integers: "))
```

13. Write a Python program to generate an array of evenly spaced numbers over a specified range using linspace.

```python
import numpy as np

def generate_evenly_spaced_numbers(start, stop, num):
    evenly_spaced_array = np.linspace(start, stop, num)
    return evenly_spaced_array

# Test the function
start = float(input("Enter the start value: "))
stop = float(input("Enter the stop value: "))
num = int(input("Enter the number of evenly spaced numbers: "))

evenly_spaced_numbers = generate_evenly_spaced_numbers(start, stop, num)
print("Array of evenly spaced numbers over the specified range:\n", evenly_spaced_numbers)
```

14. Write a program to generate an array of 10 equally spaced values between 1 and 100 using linspace.

```python
import numpy as np

# Generate array of 10 equally spaced values between 1 and 100
values = np.linspace(1, 100, 10)

# Print the generated array
print("Array of 10 equally spaced values between 1 and 100:")
print(values)
```

15. Write a Python program to create an array containing even numbers from 2 to 20 using arrange.

```python
import numpy as np

# Create array containing even numbers from 2 to 20 using arange
even_numbers = np.arange(2, 21, 2)

# Print the array
print("Array containing even numbers from 2 to 20:")
print(even_numbers)
```

16. Write a program to create an array containing numbers from 1 to 10 with a step size of 0.5 using arange.

```python
import numpy as np

# Create array containing numbers from 1 to 10 with a step size of 0.5 using arange
numbers = np.arange(1, 10.5, 0.5)

# Print the array
print("Array containing numbers from 1 to 10 with a step size of 0.5:")
print(numbers)
```