

# 1. Calculate the area of a rectangle:

```
length = float(input("Enter length of the rectangle: "))
width = float(input("Enter width of the rectangle: "))
area = length * width
print("The area of rectangle is:", area)
```

# 2. Convert miles to kilometers:

```
m=float(input("Enter distance in miles: "))
k= m*1.60934
print("The distance in kilometers is: ",k)
```

# 3. Check if a string is a palindrome:

```
def is_palindrome(s):
    return s==s[::-1]
str=input("Enter a string: ")
if is_palindrome(str):
    print("The string is a palindrome.")
else:
    print("The string is not a palindrome.")
```

# 4. Find the second largest element in a list:

```
my_list = [5, 3, 8, 1, 9, 4, 7]
second_largest = sorted(my_list)[-2]
print("The second largest element is:", second_largest)
```

# 5. indentation means in Python

Indentation is used to define the structure and hierarchy of code blocks in Python, such as loops, conditional statements, and function definitions.

# 6. Perform set difference operation:

```
set1 = {1, 2, 3, 4, 5}
set2 = {4, 5, 6, 7, 8}
difference = set1 - set2
print("The set difference is:", difference)
```

# 7. Print numbers from 1 to 10 using a while loop:

```
n = 1
```

```
while n <= 10:
    print(n)
    n += 1
```

# 8. Calculate the factorial of a number using a while loop:

```
n = int(input("Enter a number: "))
fact = 1
while n > 0:
    fact *= n
    n -= 1
print("The factorial is:", fact)
```

9. Check if a number is positive, negative, or zero using if-elif-else statements:

```
n = float(input("Enter a number: "))
if n > 0:
    print("The given number is positive.")
elif n < 0:
    print("The given number is negative.")
else:
    print("The given number is zero.")
```

# 10. Determine the largest among three numbers using conditional statements:

```
n1 = float(input("Enter the first number: "))
n2 = float(input("Enter the second number: "))
n3 = float(input("Enter the third number: "))
largest_num = n1
if n2 > largest:
    largest = n2
if n3 > largest:
    largest_num = n3
print("The largest number is:", largest_num)
```

# 11. Create a NumPy array filled with ones of given shape:

```
import numpy as np
shape = tuple(map(int, input("Enter the shape of the array: ").split()))
arr_ones = np.ones(shape)
print("Numpy array filled with ones:")
print(arr_ones)
```

# 12. Create a 2D NumPy array initialized with random integers:

```
import numpy as np
rows = int(input("Enter number of rows: "))
cols = int(input("Enter number of columns: "))
random_arr = np.random.randint(1, 100, size=(rows, cols))
print("2D Array initialized with random integers:")
print(random_arr)
```

# 13. Generate an array of evenly spaced numbers over a specified range using linspace:

```
import numpy as np
s = float(input("Enter start value: "))
e = float(input("Enter end value: "))
no_points = int(input("Enter number of points: "))
result_arr = np.linspace(s, e, no_points)
print("array of evenly spaced numbers:")
print(result_arr)
```

# 14. Generate an array of 10 equally spaced values between 1 and 100 using linspace:

```
import numpy as np
result_arr = np.linspace(1, 100, 10)
print("Array of 10 equally spaced numbers between 1 and 100: ")
print(result_arr)
```

# 15. Create an array containing even numbers from 2 to 20 using arange:

```
import numpy as np
even_arr = np.arange(2, 21, 2)
print("Array containing even numbers from 2 to 20:")
print(even_arr)
```

# 16. Create an array containing numbers from 1 to 10 with a step size of 0.5 using arange:

```
import numpy as np
arr = np.arange(1, 10.5, 0.5)
print("Array containing numbers from 1 to 10 with a step size of 0.5:")
print(arr)
```