1. Calculate the area of a rectangle:

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

2. Convert miles to kilometers:

```
miles = float(input("Enter distance in miles: "))
kilometers = miles * 1.60934
print("Distance in kilometers:", kilometers)
```

3. Check if a string is a palindrome:

```
def is_palindrome(s):
    return s == s[::-1]
string = input("Enter a string: ")
if is_palindrome(string):
    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 = [3, 7, 1, 9, 5, 4]
sorted_list = sorted(my_list)
second_largest = sorted_list[-2]
print("The second largest element is:", second_largest)
```

5. Indentation in Python:

Indentation in Python is used to define the structure of the code. It is crucial for defining blocks of code like loops, functions, and conditional statements. Incorrect indentation can lead to syntax errors.

6. Perform set difference operation:

```
set1 = {1, 2, 3, 4, 5}
set2 = {3, 4, 5, 6, 7}
difference_set = set1 - set2
print("Set difference:", difference_set)
```

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

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

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

```
num = int(input("Enter a number: "))
factorial = 1
while num > 0:
    factorial *= num
    num -= 1
print("Factorial:", factorial)
```

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

```
num = float(input("Enter a number: "))
if num > 0:
    print("Positive number")
elif num < 0:
    print("Negative number")
else:
    print("Zero")
```

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

```
num1 = 10
num2 = 20
num3 = 15
largest = max(num1, num2, num3)
print("The largest number is:", largest)
```

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

```
import numpy as np
shape = (2, 3)  # Example shape
ones_array = np.ones(shape)
print("Array filled with ones:")
print(ones_array)
```

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

```
import numpy as np
rows = 3
cols = 4
random_array = np.random.randint(1, 100, size=(rows, cols))
print("2D Array with random integers:")
print(random_array)
```

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

```
import numpy as np
start = 1
stop = 10
num_values = 5
evenly_spaced_array = np.linspace(start, stop, num=num_values)
print("Evenly spaced array:")
print(evenly_spaced_array)
```

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

```
import numpy as np
start = 1
stop = 100
num_values = 10
evenly_spaced_array = np.linspace(start, stop, num=num_values)
print("Evenly spaced array between 1 and 100:")
print(evenly_spaced_array)
```

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

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

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

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