1. Write a Python function to check whether a number is in a given range.
   Ans:

```
def in_range(number , start_number , end_number):
        return start_number <= number <= end_number;

userNumber = int(input("enter the number with range (1 to 10)"));
startNumber = 1;
endNumber = 10;

if(in_range(userNumber,startNumber,endNumber)):
        print("number is in within range (1 to 10)");
else:
        print("number is not in range (1 to 10)");
```

2. Write a Python program to get the 4 th element from front and 4 th element from last of a tuple.
   Ans:

```
my_tuple = ( 1 , 2 , 3 , 4 , 5 , 6 )
fourth_element_from_front = my_tuple[3]
tuple_length = len(my_tuple)
print("tuple length :", tuple_length)
last_element = my_tuple[-4]
print("fourth element from the front :",fourth_element_from_front)
print("fourth element from the last :",last_element)
```

3. Write a Python program to find the repeated items of a tuple
   Ans:

```
def find_repeated_items(tup):
    repeated_items = {}
    for item in tup:
        if tup.count(item) > 1:
            repeated_items[item] = tup.count(item)
    return repeated_items

# Example usage
my_tuple = (1, 2, 2, 3, 4, 4, 5, 2, 6, 6)
result = find_repeated_items(my_tuple)
if result:
    print("Repeated items in the tuple:")
    for item, count in result.items():
        print(f"{item}: {count} times")
else:
    print("No repeated items found in the tuple.")
```

4. Write a Python program to sort the tuple T= (4,2,6.8,1.8,10)
   Ans:

```
tp = (4,2,6.8,1.8,10);
print(sorted(tp))
```

5. Write a Python program to print the set difference and a symmetric difference of two sets.
   Ans:

```
def set_operations(set1, set2):
    difference = set1 - set2
    symmetric_difference = set1 ^ set2
    return difference, symmetric_difference

set1 = {1, 2, 3, 4, 5}
set2 = {3, 4, 5, 6, 7}
difference, symmetric_difference = set_operations(set1, set2)
print("Set difference (elements in set1 but not in set2):", difference)
print("Symmetric difference (elements in either set1 or set2 but not in both):",
symmetric_difference)
```

6. Write a Python program to find maximum and the minimum value in a set.
   Ans:

```
setVal = {22,45,67,89,86};
print("maximum Number: ", max(setVal))
print("minimum Number: ", min(setVal))
```

7. Write a Python program to print average of all elements of sets.
   Ans:

```
sets = {22,33,55,66,77}
avg = sum(sets)/len(sets);
print(avg)
```

8. Write a Python program to create a tuple using two different tuples.
Ans:

```
tp1 = (2,3,4,5)
tp2 = (77,99,25,79)
tp3 = tp1 + tp2;
print(tp3)
```

9. Write an anonymous function to calculate area of square
Ans:

```
n = int(input("Enter any number "));

areaSquare = lambda side:side**2

print(f"{n} Area of square is {areaSquare(n)}");
```

10. Write a python program to accept the string which containing all vowels
Ans:

```
string = input("Enter a string: ")

vowels = "aeiouAEIOU"

print("Vowels in the string:", end=" ")
for char in string:
   if char in vowels:
      print(char, end=" ")
```

11. Write a Python program to find the length of a string without using built-in function.
Ans:

```
def string_length(string):
   count = 0
   for char in string:
    count += 1
   return count

my_string = "Hello, world!"
length = string_length(my_string)
print(f"The length of the string '{my_string}' is {length}")
```

12. Write a program which prints Fibonacci series of a number
Ans:

```
def fibo(n):
   if(n==0 or n==1):
      return 1;
   else:
      return (fibo(n-1) + fibo(n-2));

num = int(input("Enter the fibonacci series index number "));

for numbers in range(0, num):
   print(f"{fibo(numbers)}  ", end="")
```

13. Write a Python program to create a tuple of n numbers and print maximum, minimum, and sum of elements in a tuple.
Ans:

```
nTp = (12,354,5,642,5,53,2,6,764);
maxNumber = max(nTp);
minNumber = min(nTp);
sumNTp = sum(nTp);

print("Max: ", maxNumber)
print("Min: ", minNumber)
print("Sum: ", sumNTp)
```

14. Write a python program to check if a string is a Palindrome or not
Ans:

```
string = input("Enter any string ");

originalString = string;

reversedString = string[::-1]

print(f"Original String is: {originalString}")
print(f"reversed String is: {reversedString}")

if(originalString == reversedString):
     print("Given string is palindrome ")
else:
     print("Given string is not a palindrome ")
```

15. Write python program to create a tuple of n number print the first half value of tuple in one line and last half values of tuple on next line

Ans:

```
def find_repeated_items(tup):
    repeated_items = {}
    for item in tup:
        if tup.count(item) > 1:
            repeated_items[item] = tup.count(item)
    return repeated_items

# Example usage
my_tuple = (1, 2, 2, 3, 4, 4, 5, 2, 6, 6)
result = find_repeated_items(my_tuple)
if result:
    print("Repeated items in the tuple:")
    for item, count in result.items():
        print(f"{item}: {count} times")
else:
    print("No repeated items found in the tuple.")
```

16. Write a Python program to accept and convert string in uppercase or vice versa

Ans:

```
string = input("Enter any string ");
print("Original string: ", string.upper());
print("Upper case  string: ", string.upper());
```

17. Write a Python program to calculate the average of numbers in a given list.

Ans:

```
lst = [88, 98, 79, 89, 56];
avg = sum(lst)/len(lst);

print("List: ", lst);
print("Average of given list : ", avg);
```

18. Write a Python program to unpack a tuple in several variables

Ans:

```
tp = (1, 2, 3, 4, 5)

var1, var2, var3, var4, var5 = tp

print("Var1:", var1)
print("Var2:", var2)
print("Var3:", var3)
print("Var4:", var4)
print("Var5:", var5)
```