Accept a string and print it in reverse using slicing

```python
s = input()

print(s[::-1])
```

Accept a string and replace all vowels with a *

```python
s = input()

print(''.join(['*' if i in 'aeiouAEIOU' else i for i in s]))
```

Create a dictionary with 5 elements where the value of each is the squared value of the key

```python
d = {i: i*i for i in range(1, 6)}

print(d)
```

Create a tuple and demonstrate unpacking it into variables

```python
t = (1, 2, 3)

a, b, c = t

print(a, b, c)
```

Write a program to create a new text file and write your name into it

```python
f = open("newfile.txt", "w")

f.write("YourName")

f.close()
```

Write a program to append user input to an existing file

```python
f = open("newfile.txt", "a")

f.write(input())

f.close()
```

Create a program that reads a file line by line and prints each line

```python
f = open("newfile.txt", "r")

for line in f:

    print(line, end="")

f.close()
```

Write a program to count the number of lines in a file

```python
f = open("newfile.txt", "r")

print(len(f.readlines()))

f.close()
```

Write a program to count the total number of words in a file

```python
f = open("newfile.txt", "r")

print(len(f.read().split()))

f.close()
```

Write a program that overwrites a file with new content

```python
f = open("newfile.txt", "w")

f.write("New content")

f.close()
```

Check if a file exists before reading or writing using the os module

```python
import os

if os.path.exists("newfile.txt"):

    f = open("newfile.txt", "r")

    print(f.read())
```

```
        f.close()

else:

    print("File does not exist")
```

Create a class Person with attributes name and age. Write a method to display the person's details

```
class Person:

    name = "Alice"

    age = 30

    display = print(name, age)
```

Write a class Rectangle with methods to compute area and perimeter

```
class Rectangle:

    l = 5

    b = 3

    area = l * b

    perimeter = 2 * (l + b)

    print(area)

    print(perimeter)
```

Define a class Circle with a method to calculate area. Accept radius as input from the user

```
class Circle:

    r = float(input())

    area = 3.14159 * r * r

    print(area)
```

Define a class Book with attributes title, author, and price. Write a method to display

book details

```python
class Book:

    title = "Sample Book"

    author = "Author Name"

    price = 299

    print(title, author, price)
```


Create a base class Animal and a subclass Dog that adds a method bark()

```python
class Animal:

    pass

class Dog(Animal):

    print("Bark")
```


Create a class BankAccount with attributes account_holder and balance. Initialize them

through a constructor

```python
class BankAccount:

    def __init__(self):

        self.account_holder = "John"

        self.balance = 1000

        print(self.account_holder, self.balance)

BankAccount()
```


Showcase the use of the super() method in OOPs by incorporating in a program

```python
class Parent:

    def __init__(self):

        print("Parent")
```

```python
class Child(Parent):

    def __init__(self):

        super().__init__()

        print("Child")

Child()
```

Write a class Patient that stores name and disease. Use methods to access and update data securely

```python
class Patient:

    __name = "John"

    __disease = "Flu"

    print(__name, __disease)
```