

12. Create a `Movie` class with attributes for title, director, and release year. Add a method to display the movie's information. Create an object and call the method.

```
class Movie:
    def __init__(self, title, director, release_year):
        self.title = title
        self.director = director
        self.release_year = release_year
    def movie_information(self):
        print("Title:", self.title)
        print("Director:", self.director)
        print("Release Year:", self.release_year)

movie = Movie("Ready", "Navdeep Singh", 2024)
movie.movie_information()
```

```
➞ Title: Ready
   Director: Navdeep Singh
   Release Year: 2024
```

13. Define a `Product` class with attributes for name and price. Add a method to calculate the price after tax (assume a fixed tax rate). Create an object and print the price after tax.

```
class Product:
    def __init__(self, name, price):
        self.name = name
        self.price = price

    def calculate_price_after_tax(self):
        tax_rate = 0.10
        tax_amount = self.price * tax_rate
        total_price = self.price + tax_amount
        return total_price

product = Product("Laptop", 50000)
price_after_tax = product.calculate_price_after_tax()
print("Price after tax:", price_after_tax)
```

```
➞ Price after tax: 55000.0
```

14. Create a `Player` class with attributes for name and score. Add a method to update the score. Create an object and test the method.

```
class Player:
    def __init__(self, name, score):
        self.name = name
        self.score = score

    def update_score(self, new_score):
        self.score = new_score

player = Player("Navdeep Singh", 100)
print("Initial Score:", player.score)

player.update_score(200)
print("Updated Score:", player.score)
```

```
➞ Initial Score: 100
   Updated Score: 200
```

15. Define a `House` class with attributes for address and number of rooms. Add a method to display the house's details. Create an object and call the method.

```
class House:
    def __init__(self, address, number_of_rooms):
        self.address = address
        self.number_of_rooms = number_of_rooms
    def house_detail(self):
        print("Address:", self.address)
        print("Number of Rooms:", self.number_of_rooms)

house = House("vill.Rasulra", 5)
house.house_detail()
```

```
➞ Address: vill.Rasulra
   Number of Rooms: 5
```

16. Create a Shape class with attributes for color and filled (a boolean). Add a method to display the shape's properties. Create an object and call the method.

```
class Shape:
    def __init__(self, color, filled):
        self.color = color
        self.filled = filled
    def shape_properties(self):
        print("Color:", self.color)
        print("Filled:", self.filled)
shape = Shape("blue", "black")
shape.shape_properties()
```

```
➞ Color: blue
   Filled: black
```

17. Define a Vehicle class with attributes for type and speed. Add a method to accelerate (increase the speed). Create an object and test the method.

```
class Vehicle:
    def __init__(self, type, speed):
        self.type = type
        self.speed = speed

    def accelerate(self, acceleration):
        self.speed += acceleration

vehicle = Vehicle("Car", 0)
print("Initial Speed:", vehicle.speed)

vehicle.accelerate(50)
print("Accelerated Speed:", vehicle.speed)
```

```
➞ Initial Speed: 0
   Accelerated Speed: 50
```

18. Create a Country class with attributes for name and population. Add a method to display the country's information. Create an object and call the method.

```
class Country:
    def __init__(self, name, population):
        self.name = name
        self.population = population
    def country_information(self):
        print("Name:", self.name)
        print("Population:", self.population)
country = Country("India", "250 million")
country.country_information()
```


```
➞ Name: India
   Population: 250 million
```

19. Define a Student class with attributes for name and a dictionary of subjects and grades. Add a method to calculate the average grade for all subjects. Create an object and print the average grade.

```
class Student:
    def __init__(self, name, subjects, grades):
        self.name = name
        self.subjects = subjects
        self.grades = grades

    def average_grade(self):
        total_grade = sum(self.grades)
        average = total_grade / len(self.grades)
        return average


student = Student("Navdeep Singh", ["Maths", "Science"], [85, 90, 75])
print(student.average_grade())
```

 83.33333333333333

20. Create a Calendar class with attributes for day, month, and year. Add a method to display the date in DD/MM/YYYY format. Create an object and call the method.

```
class Calender:
    def __init__(self, day, month, year):
        self.day = day
        self.month = month
        self.year = year
    def Display_Date(self):
        print("Date:",self.day)
        print("Month:",self.month)
        print("Year:",self.year)

calender = Calender(2,5,2005)
calender.Display_Date()
```

 Date: 2
Month: 5
Year: 2005