1. Design an encapsulated class for UML diagram

Telegram
-name:String -contact_no:long -email:String
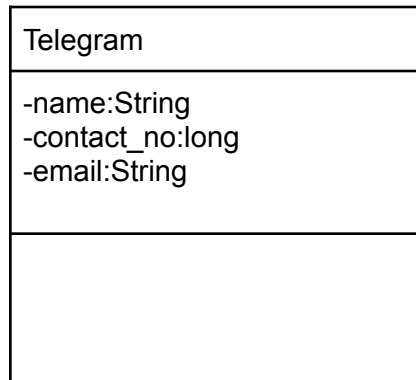
1. Create a class that corresponds to this UML diagram, with all data members as private and getters and setters defined for each one and Create a parameterized constructor to initialize the values

.

2 . Create a Driver class
- Create an object for the class telegram
- Initialize the value "Tom" to name ,9909873471 to contact_no,"Tom12@gmail.com" to Email.
- Display the name without using getter method

- Create an object for the class telegram
- Initialize the value "Jerry" to name ,9909201471 to contact_no,"Jerry124@gmail.com" to Email.
- Display the contact without using getter method

2.Create a class that corresponds to this uml diagram,with all the data members as private,Declare a parameterized constructor to initialize the value

| Instagram |
|---|
| -userName:String
-pwd:String
-email:String |
| |
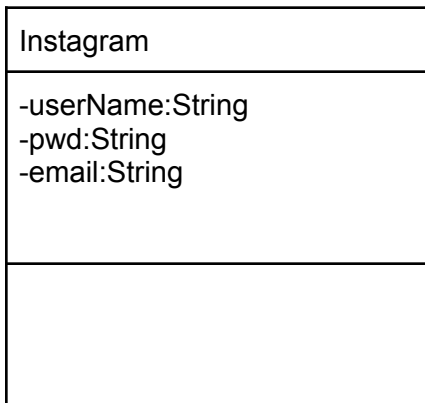
1.Create a Driver class

- Create an object for the class Instagram
- Initialize the value "John" to name ,"Tiger" to pwd,"John12@gmail.com" to Email.
- Display the userName

- Create an object for the class Instagram
- Initialize the value "John" to name ,"Tiger" to pwd,"John12@gmail.com" to Email.
- Display the pwd

3. Design an encapsulated class for the following UML diagram
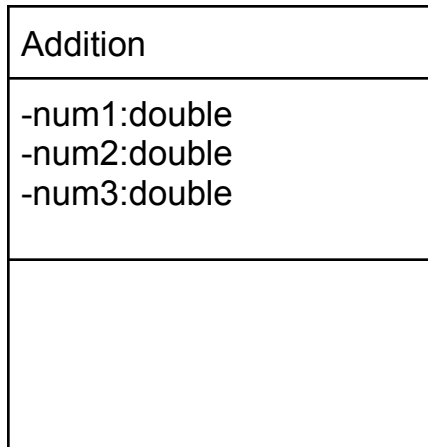
Addition
-num1:double -num2:double -num3:double

- ● Create an object for the class Addition
- ● Initialize the value 10 to num1 ,50 to num2 ,100 to num3.
- ● Fetch the values from the addition object and perform addition of three numbers.
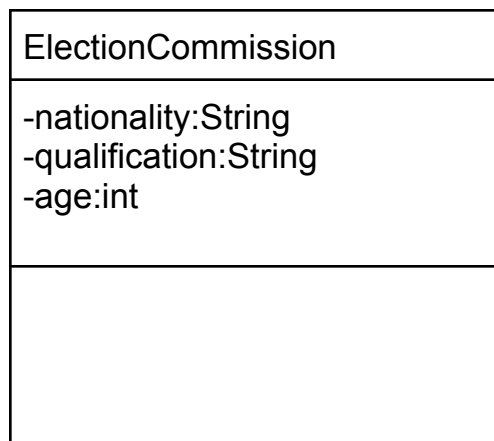
- ● Create an object for the class Addition
- ● Initialize the value 10 to num1 ,50 to num2 ,100 to num3.
- ● Fetch the values from the addition object and perform addition of three numbers.

4. Design an encapsulated class for the following UML diagram.
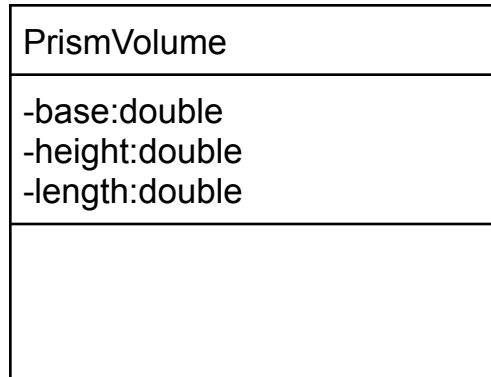
PrismVolume
-base:double -height:double -length:double

- Create an object for the class PrismVolume
- Initialize the value 40 to base ,10 to height ,200 to length.
- Fetch the values from the Prism Object and find the volume of the Prism. Volume of prism(V=BH).

- Create an object for the class PrismVolume
- Initialize the value 40 to base ,10 to height ,200 to length.
- Fetch the values from the Prism Object and find the volume of the Prism.

  .

5.Create a class that corresponds to this uml diagram,with all the data members as public,Declare a parameterized constructor to initialize the value

ElectionCommission
-nationality:String -qualification:String -age:int

Case-01—->Negative scenario
- Create a parameterized constructor and initialize the values
- Create a Driver class
- Create an object for the ElectionCommission class
- Fetch the values present in ElectionCommission Object
- Compare the values with the standard values(nationality!= Indian qualification!=graduate,age<25) and check whether the candidate is eligible to participate in elections or not..

Case-02—------------>Positive scenario
- Create a parameterized constructor and initialize the values
- Create an object for the ElectionCommission class
- Fetch the values present in ElectionCommission Object
- Compare the values with the standard values(nationality=indian,qualification=graduate,age>=25) and check whether the candidate is eligible for participating in elections.

6. Create a class that corresponds to this UML diagram, with all data members as private and getters and setters defined for each one and Create a parameterized constructor to initialize the values

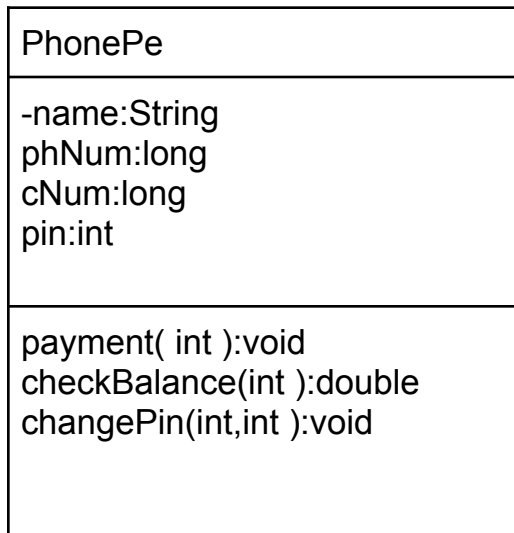| Matrimony |
| --- |
| -name:String
-age:int
-gender:String |
| |
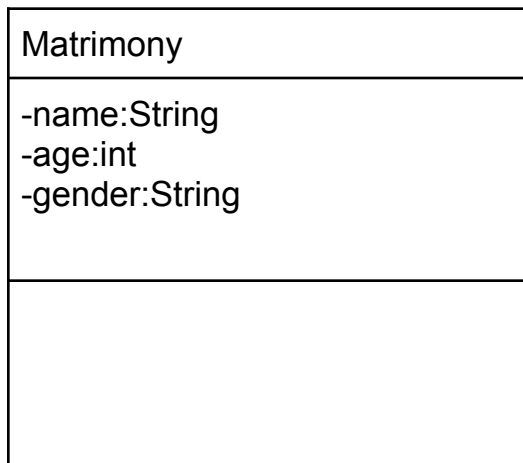
- Create a parameterized constructor and initialize the values
- Create a Driver class
- Create multiple objects for the class Matrimony
- Fetch the genders of different objects and compare them,if genders are not equal ,then compare the ages.
- If both the conditions are satisfied print a statement saying "You Found Your Match".

7. Create a class that corresponds to this UML diagram, with all data members as private and getters and setters defined for each one and Create a parameterized constructor to initialize the values

.

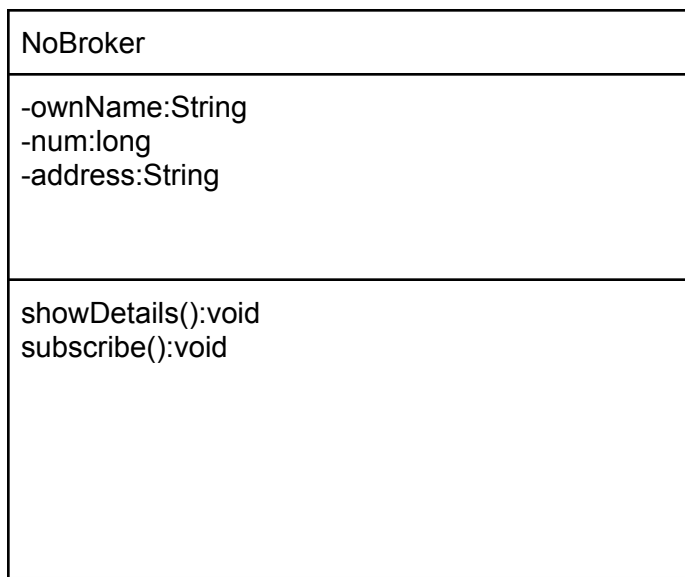| Games |
|---|
| -playerName:String
-pAge:int
-sportsName:String |
| details(String,String):boolean |
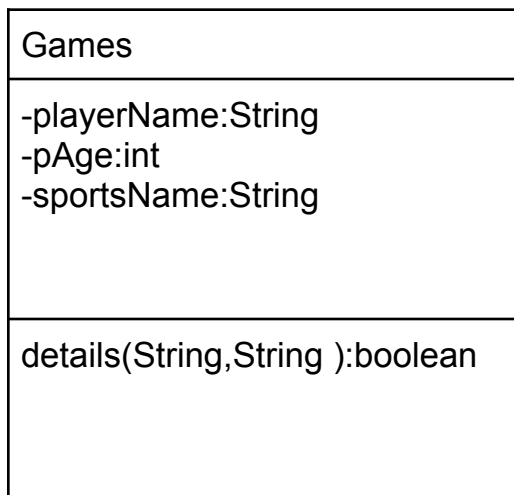
- Create a parameterized constructor and initialize the values
- Create a driver class
- Create multiple objects for Games class
- Fetch the sportsName of different object and pass the value as an argument for details method
- In details method Compare both the sports ,if both are playing same game.
- If the details method returns true print the details of the players

8. Create a class that corresponds to this UML diagram, with all data members as private and getters and setters defined for each one and Create a parameterized constructor to initialize the values

| PhonePe |
| --- |
| -name:String
phNum:long
cNum:long
pin:int |
| payment(int):void
checkBalance(int):double
changePin(int,int):void |

- Create a parameterized constructor and initialize the values
- Create a driver class
- Create an object for Phonepay
- In payment method,validate the pin entered by the user
- If the validation is successful print appropriate message
- In checkBalance method,validate the pin entered by the user
- In changePin method,accept current pin and new pin,validate the pin and update.

9. Create a class that corresponds to this UML diagram, with all data members as private and getters and setters defined for each one and Create a parameterized constructor to initialize the values

NoBroker
-ownName:String -num:long -address:String
showDetails():void subscribe():void

- Create a parameterized constructor and initialize the values
- Create a method subscribe in noBroker class and internally call show details method
- Create a Driver class
- Create an object for noBroker class
- Call the subscribe method

10. Create a class that corresponds to this UML diagram, with all data members as private and getters and setters defined for each one and Create a parameterized constructor to initialize the values

Fitness
name:String num:long
bookingDetails();void checkIn(String,long);void
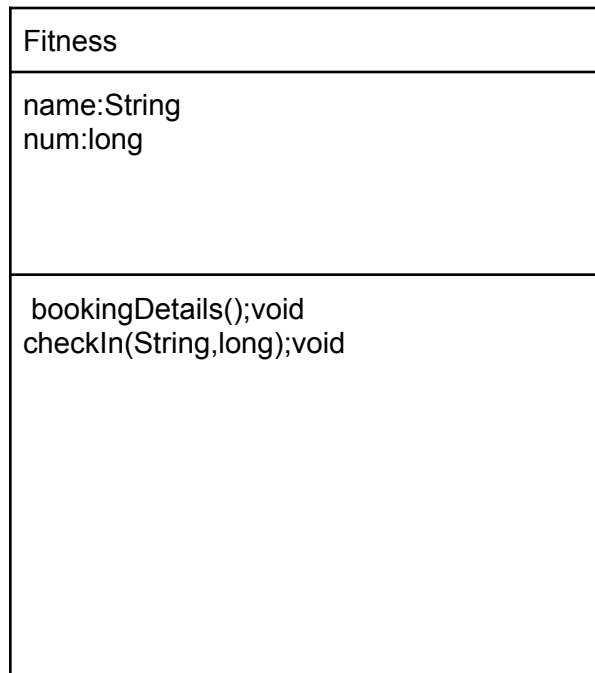
- Create a parameterized constructor and initialize the values
- Create a method bookingDetails which displays the details and a message Stating "Your class is booked".
- Create a checkIn() which should validate name and number
  If it is valid the call bookingDetails method
- Create a driver class
- Create an object of Fitness class and call checkIn().