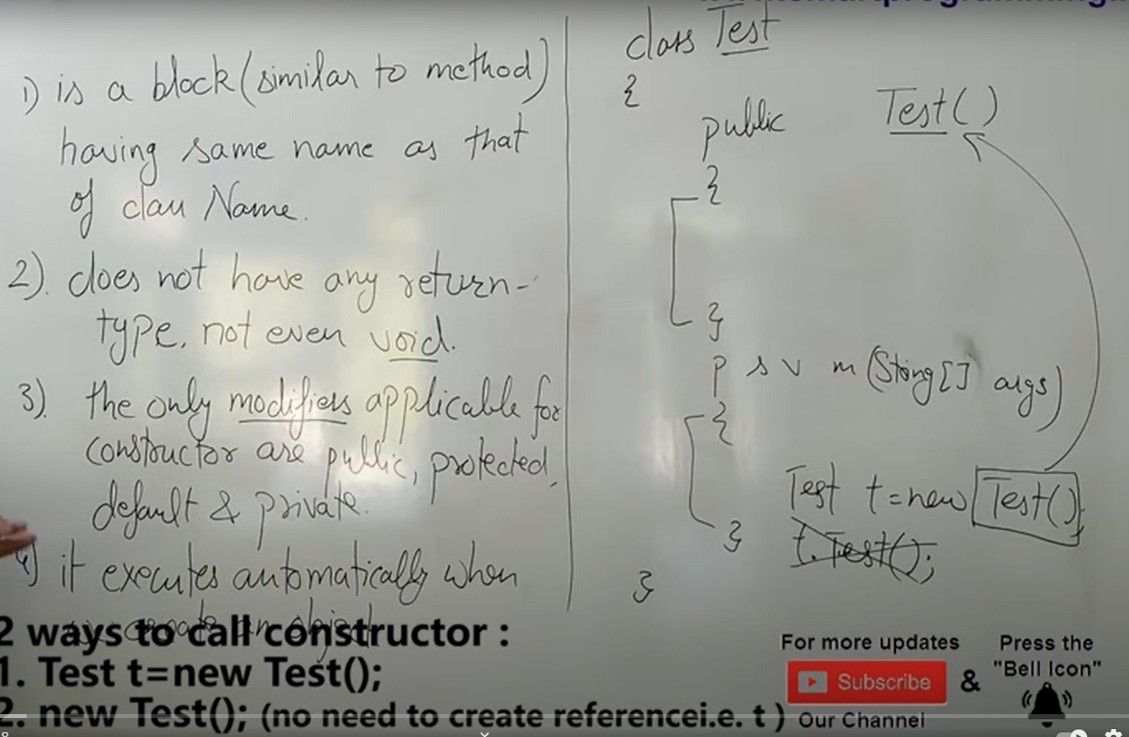
*Constructor*

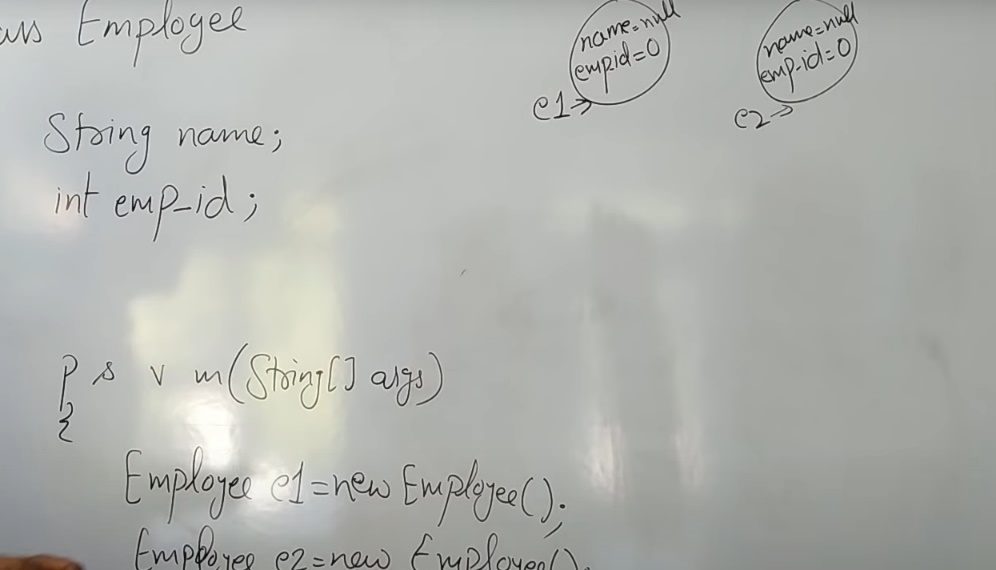
***Constructor starts execute automatically when we create object for constructor. Here no need to create object for method.***

**

***Use of constructor***

***Here creating object e1 and e2 for 2 instance variables name and emp\_id and we are not initializing any value.so will take default values.***

***Its not a good way to write program…lot of objects we can create like this. Here two variables same value.. it will create problems in project.***

******

***3 ways to create objects.***

***By using reference variable***

***By using method***

***By using constructor***

***By using reference variable if we initialize and create objects, again we will be getting same values for objects. So this also not a good idea***

***Class Employee{***

***String Name=”abc”; e1name=abc;emp\_id=101 e2name=abc;emp\_id=101***

***Int emp\_id=101;***

***}***

***Psvm(String[] args);{***

***Employee e1=new Employee();***

***Employee e2=new Employee ();***

***}***

***Another way***

***Initializing values separately for each objects. Here the problem is when we have 1000+ emplids then we have to create that much objects and values so no use.***

***Class Employee{***

***String Name;***

***Int emp\_id;***

***psvm(String[] args){***

***Employee e1=new Employee();***

***e1.Name=”abc”;***

***e1.emp\_id=101;***

***Employeee2=new Employee();***

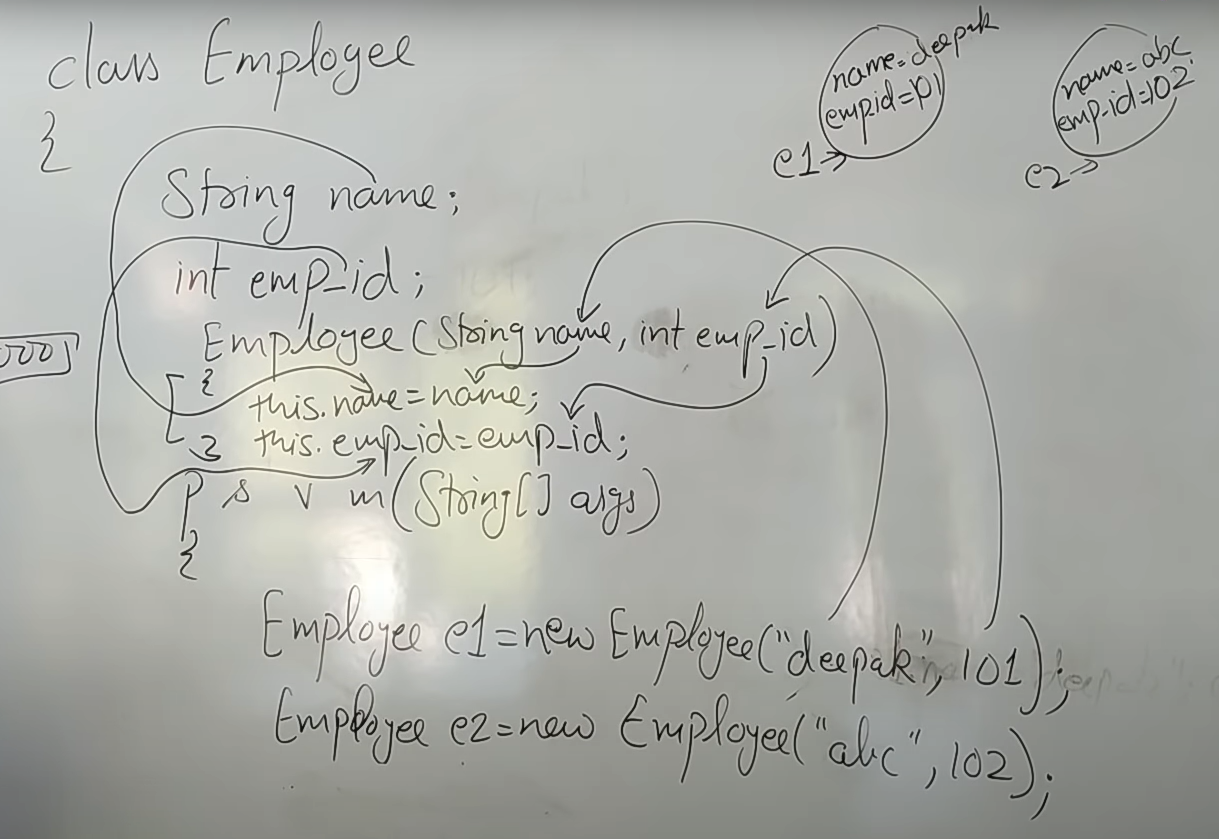
***e2.Name=”ddd”;***

***e2.emp\_id=202;***

***}***

***}***

***We need unique values for each objects so we use constructor for this class with the parameterized constructor.***

******

***Class Employee{***

***String Name;***

***int emp\_id;***

***public Employee(String Name, int emp\_id);***

***{***

***this.name=name;***

***this.emp\_id= emp\_id;***

***}***

***psvm(String[] args);{***

***Employee e1=new Employee(“abc”,111); //here unique values get initialize inside objects***

***Employee e2=new Employee(“ddd”,102); // To initialize an object, and no need to create object here***

***}***

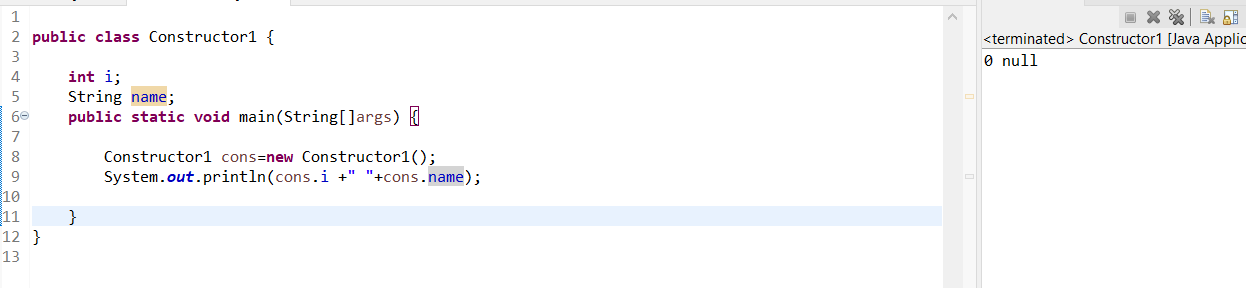
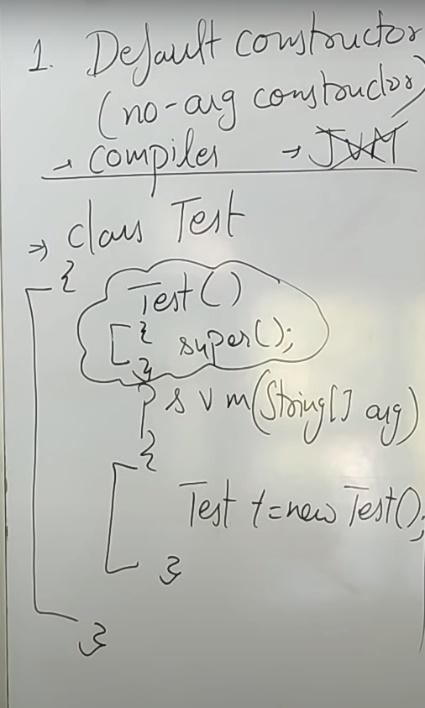
***}***

***3types of constructor***

* ***Default constructor (No argument constructor)***
* ***No arg constructor(User defined constructor)***
* ***Parameterized Constructor***
* ***Default constructor (No argument constructor)***

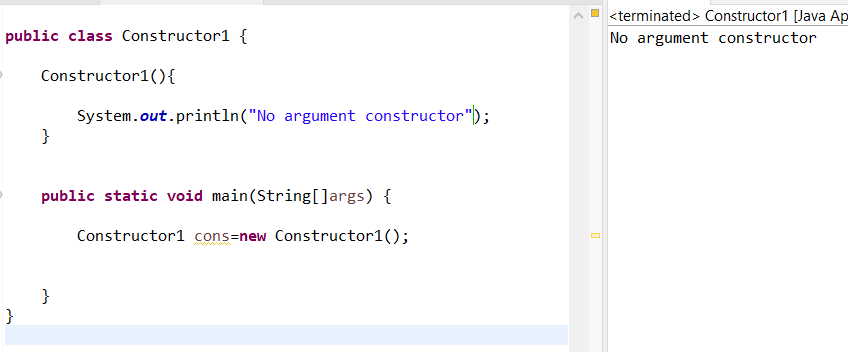
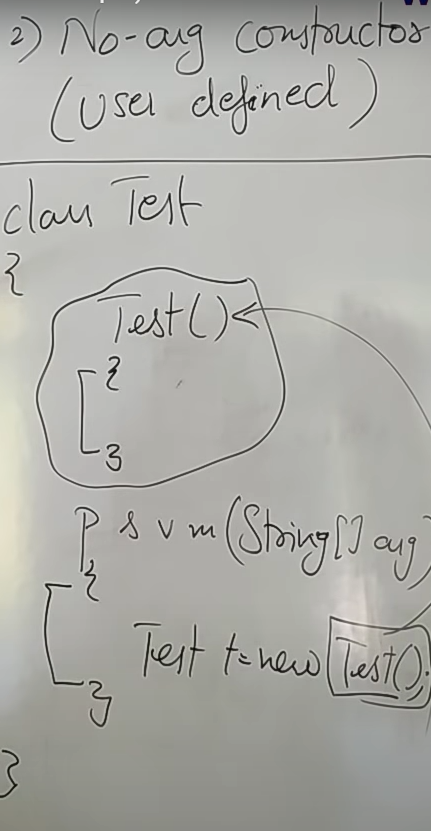
***Default constructor is created by compiler not JVM with no arguments or no parameters***

***When we create object ,compiler will create default constructor. Here compiler created default constructor and initialized default values also. Compiler wont judge return type***

******

* ***No arg constructor(User defined constructor)***

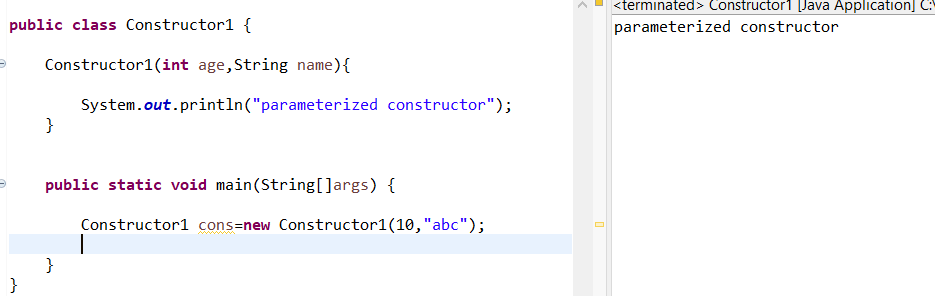
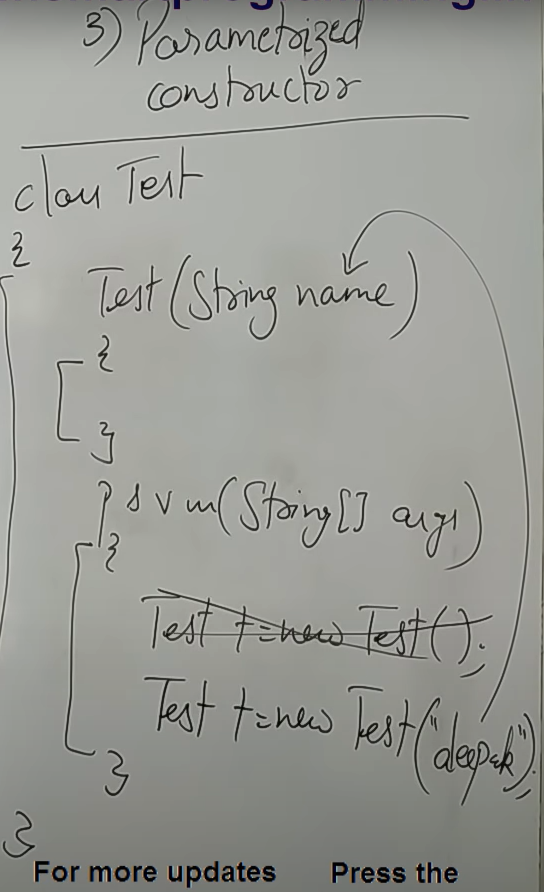
***Here programmer/user will create constructor with no arguments. And compiler wont create any default constructor here***

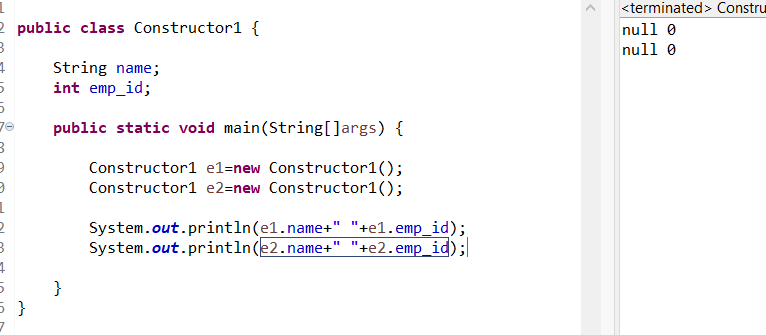
******

* ***Parameterized Constructor***

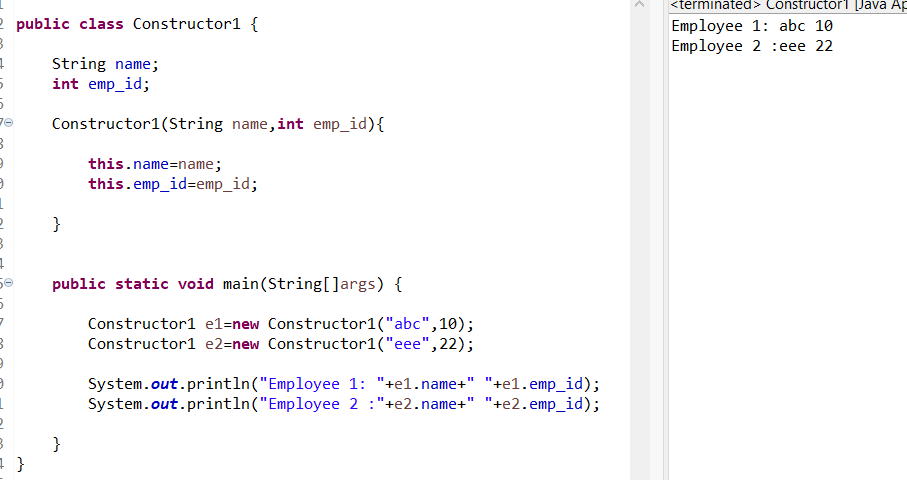
***Here programmer/user will create constructor with parameters and compiler wont create any default constructor here***

***Initializing values through objects.no use of return types in constructor since we are initializing through objects.***

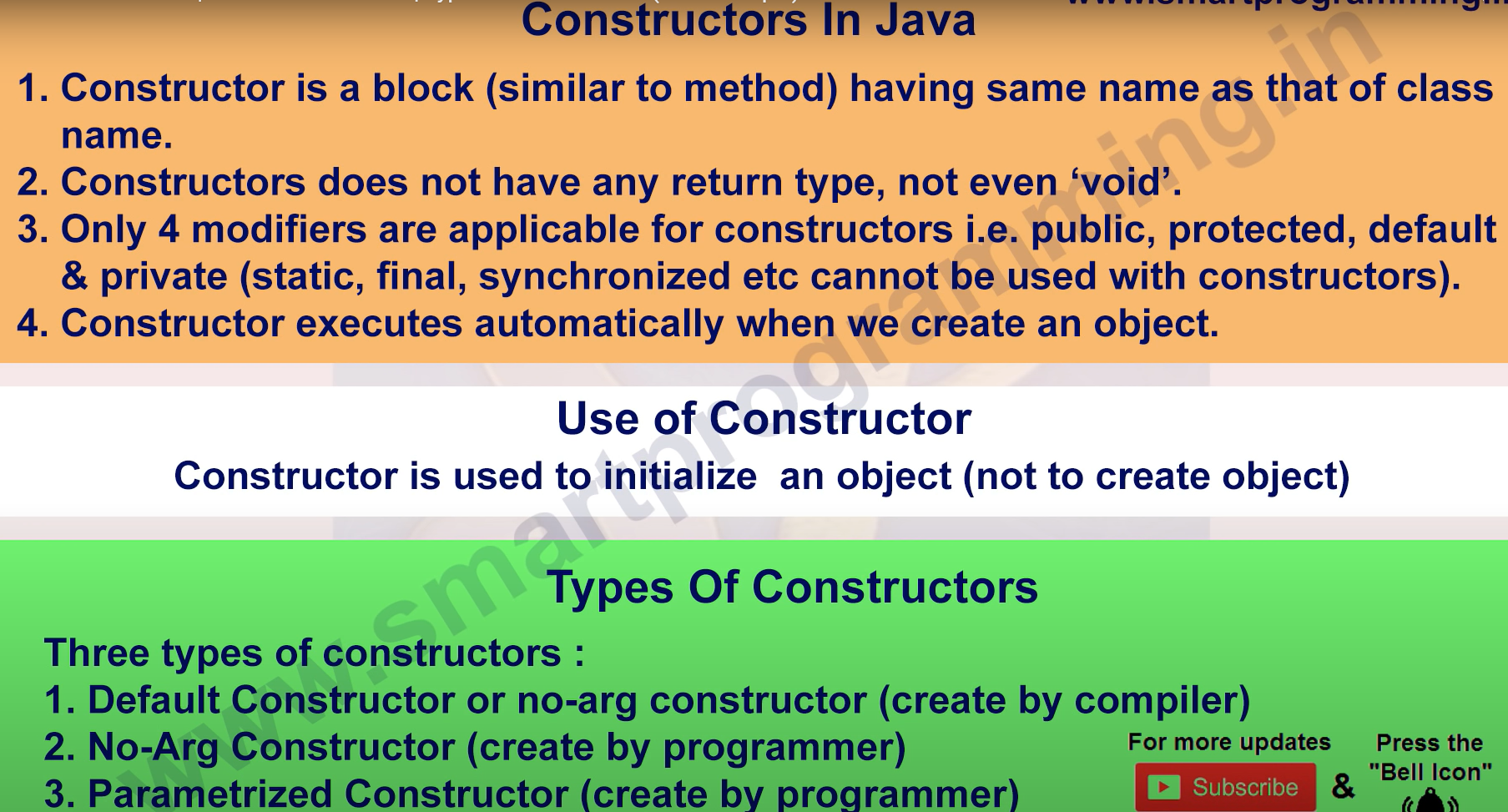
******

******

***So to get unique values for objects we are using constructor. Initialize values to objects via parameterized method.***

******

***Notes***

******