Java Training Index

Day1: (5-08-24)

Java Basics:

1. What is Computer
2. Flow Diagram
3. Languages and Applications
4. Why Java
5. Java Features
6. Jvm,Jre,Jdk
7. Operators, datatypes,variables,keywords,loops,Conditional Statements
8. Basic Programs- BigTwo, BigThree, BigFive, Swap

Day2:(6-08-24)

1. Nested Loops
2. Arrays- 1D,2D
3. Packages
4. Logical Programs- nested loop progs,array progs, pattern progs,swap,Factorial,Fibonacci
5. Scanner class
6. Enums- Enumeration
   : Example Program
7. Event management Application- understanding and Analysing

Day3:(7-08-24)

1. OOPS: Encapsulation
   Programs: Calculation, Person, MethodsFlow

```
CalculationD... | *HAS_A_DEMO.... | StaticDemo1.... | StaticDemo2.... | StaticDemo3.... | StaticDemo4.... | StaticD
1  package com.evergent.corejava.oops;
2
3  public class HAS_A_DEMO {
4      public void myData()
5      {
6          System.out.println("HAS_A_DEMO");
7      }
8  // we can call method of anaother class without using extends
9  // we create an object of another class and call it without using extends
10     public static void main(String[] args) {
11         // TODO Auto-generated method stub
12         HAS_A_DEMO has= new HAS_A_DEMO();
13         has.myData();
14         Person p=new Person();
15         p.display();
16
17     }
18
19 }
20
Problems | Javadoc | Declaration | Console
<terminated> CalculationDemo [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023-03\plugins\org.eclip
30
-10
200
0
```

```
CalculationD... | StaticDemo1.... | StaticDemo2.... | StaticDe
1  package com.evergent.corejava.oops;
2
3  public class CalculationDemo {
4      int a=10,b=20;
5      int c;
6      public void addition()
7      {
8          c=a+b;
9          System.out.println(c);
10     }
11     public void subtraction()
12     {
13         c=a-b;
14         System.out.println(c);
15     }
16     public void multiplication()
17     {
18         c=a*b;
19         System.out.println(c);
20     }
21     public void divisipn()
22     {
23         c=a/b;
24         System.out.println(c);
25     }
26
27     public static void main(String[] args) {
28         // TODO Auto-generated method stub
29         CalculationDemo cd= new CalculationDemo();
30         cd.addition();
31         cd.subtraction();
32         cd.multiplication();
33         cd.division();
34
35
36     }
37
38 }
39
Problems | Javadoc | Declaration | Console
<terminated> CalculationDemo [Java Application] C:\Users\dhruti.guthiko
30
-10
200
0
```

```
package com.evergent.corejava.oops;

public class MethodsFlow {
    //No parameters with no return type
    public void show()
    {
        System.out.println("No parameters with no return type");
    }
    //parameter with no return type
    public void myData(int a,int b)
    {
        System.out.println(a+b);
    }
    //parameter with return type
    public int  myMul(int a,int b)
    {
        return a*b;
    }
    //no parameters with return type
    public int myChange()
    {
        return 100;
    }

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        MethodsFlow mf = new MethodsFlow();
        mf.show();
        mf.myData(10,5);
        System.out.println(mf.myMul(10, 5));
        System.out.println(mf.myChange());

    }
}
```

```
30
-10
200
0
```

```
package com.evergent.corejava.oops;
class MyBigData
{
    public void myData()
    {
        System.out.println("MyBigData");
    }
    public void myData1()
    {
        System.out.println("MyBigData");
    }
}

public class MethodOverriding extends MyBigData {
    public void MyData()

    {
        System.out.println("Mydata");
    }
    public void show()
    {
        System.out.println("Hello");
    }
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        MethodOverriding mo = new MethodOverriding();
        mo.myData();
        mo.myData1();
        mo.show();
    }
}
```

```
MyBigData
MyBigData
Hello
```

2. Inheritance

```
package com.evergent.corejava.oops;
class MyPerson {
    public void personInfo()
    {
        System.out.println("Dhruti");
    }
}
class PersonDetails extends MyPerson
{
    public void personData()
    {
        System.out.println("Hyderabad");
    }
}

public class MultiLevelInheritance extends PersonDetails {
    public void show()
    {
        System.out.println("Vinay");
    }
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        MultiLevelInheritance mi=new MultiLevelInheritance();
        mi.personInfo();
        mi.personData();
        mi.show();
    }
}
```

```
Dhruti
Hyderabad
Vinay
```

3. Polymorphism: Method Overloading, Method Overriding

```
package com.evergent.corejava.oops;

public class MyClass extends CalculationDemo {
    public void show()
    {
        System.out.println("Dhruti");
    }

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        MyClass mc= new MyClass();
        mc.show();
        mc.addition();

    }

}
```

```
Dhruti
30
```

```
package com.evergent.corejava.oops;

public class Person {
    String name="Dhruti";
    int age=22;
    String address="Hyderabad";
    public void display()
    {
        System.out.println("Name:"+name);
        System.out.println("Age:"+age);
        System.out.println("Address:"+address);
    }

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Person p= new Person();
        p.display();


    }

}
```

```
Name:Dhruti
Age:22
Address:Hyderabad
```

```java
package com.evergent.corejava.oops;

public class UserLogin {
    public void loginDetails()
    {
        System.out.println("login details");
    }
    public void loginDetails(String username,String pass)
    {
        System.out.println("Username:"+username);
        System.out.println("password:"+pass);
    }
    public void loginDetails(String uname,String pass,String captcha)
    {
        System.out.println("Username:"+uname);
        System.out.println("password:"+pass);
        System.out.println("Captcha:"+captcha);
    }
    public void loginDetails(int mobile,String pass)
    {
        System.out.println("Mobile:"+mobile);
        System.out.println("password:"+pass);
    }
    public void show()
    {
        System.out.println("Hello");
    }
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        UserLogin ulogin = new UserLogin();
        ulogin.loginDetails();
        ulogin.loginDetails("Dhruti","Dhru123");
        ulogin.loginDetails("Vinay","Dhru123","xyz");
        ulogin.loginDetails(1111111111,"Dhru123");
        ulogin.show();
    }
}
```

```
login details
Username:Dhruti
password:Dhru123
Username:Vinay
password:Dhru123
Captcha:xyz
Mobile:1111111111
```

4. Abstraction
5. IS-A, HAS-A
6. System.out.println

Day4:(8-08-24)

1. CONSTRUCTORS
2. This keyword
3. Super keyword
4. Copy Constructors

```java
package com.evergent.corejava.constructor;

public class Employee1 {

    Employee1() // Default constructor
    {
        System.out.println("Default Constructor");
    }

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        new Employee1();// Object Creation


    }

}
```

```
<terminated> Employee1 [Java Application] C:\Users\dhruti.guthikonda\
Default Constructor
```

```java
    public void display()//method
    {
        System.out.println("Employee number:"+eno);
        System.out.println("Employee name:"+ename);
        System.out.println("Employee salary:"+sal);
    }

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        new Employee2();// Object Creation
        Employee2 emp2=new Employee2(21,"Dhruti",600000);//object reference
        emp2.display();



    }
}
```

```
<terminated> Employee2 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\ecl
Default Constructor
Employee number:21
Employee name:Dhruti
Employee salary:600000.0
```

Person.java | MyClass.java | MultiLevell... | UserLogin.java | MethodOverr... | Employee1.java | Employe

```java
package com.evergent.corejava.constructor;

public class Employee3 {
    int eno;//default=0 , instance variables
    String ename;//null
    double sal;//0.0

    Employee3() // Default constructor
    {
        System.out.println("Default Constructor");
    }

    Employee3(int eno,String ename,double sal) // Parameterised const, eno1,ename1..-local variables
    {
        this.eno=eno;// this keyword only points to instance variables.
        this.ename=ename;
        this.sal=sal;
    }
    public void display()//method
    {
        System.out.println("Employee number:"+eno);
        System.out.println("Employee name:"+ename);
        System.out.println("Employee salary:"+sal);
    }

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        new Employee3();// Object Creation
        Employee3 emp2=new Employee3(21,"Dhruti",600000);//object reference
        emp2.display();
    }
}

```

Problems | Javadoc | Declaration | Console ×

```
<terminated> Employee3 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023-03\plugins\org.eclip
Default Constructor
Employee number:21
Employee name:Dhruti
Employee salary:600000.0
```

Person.java | MyClass.java | MultiLevell... | UserLogin.java | Employee1.java | E

```java
package com.evergent.corejava.constructor;

public class Employee4 {

     void Employee4() // it is considered as a method because of void
    {
        System.out.println("Method");
    }

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Employee4 emp4 = new Employee4();// Object Creation
        emp4.Employee4();


    }
}

```

Problems | Javadoc | Declaration | Console ×

```
<terminated> Employee4 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclip
Method
```

Person.java | MyClass.java | UserLogin.java | Employee1.java | Employee2.java

```java
package com.evergent.corejava.constructor;

public class Employee5 {
    int eno;//default=0 , instance variables
    String ename;//null
    double sal;//0.0

    Employee5() // Default constructor
    {
        System.out.println("Default Constructor");
    }
    Employee5(int eno)
    {
        this.eno=eno;
    }

    Employee5(int eno,String ename,double sal) // Parameterised const, eno1,e
    {
        this(eno);// this keyword only points to instance variables.
        this.ename=ename;
        this.sal=sal;
    }
    public void display()//method
    {
        System.out.println("Employee number:"+eno);
        System.out.println("Employee name:"+ename);
        System.out.println("Employee salary:"+sal);
    }

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        new Employee5();// Object Creation
        Employee5 emp5=new Employee5(21,"Dhruti",600000);//object reference
        emp5.display();
    }
}

```

Problems | Javadoc | Declaration | Console ×

```
<terminated> Employee5 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclip
Default Constructor
Employee number:21
Employee name:Dhruti
Employee salary:600000.0
```

```java
package com.evergent.corejava.constructor;
class MyEmployee
{
    int eno;
    public MyEmployee()
    {

    }
    MyEmployee(int eno)
    {
        this.eno=eno;
        System.out.println("Employee No super class:"+eno);
    }
}
public class Employee6 extends MyEmployee{
    int eno;//default=0 , instance variables
    String ename;//null
    double sal;//0.0
    Employee6() // Default constructor
    {
        System.out.println("Default Constructor");
    }
    Employee6(int eno)
    {
        this.eno=eno;
    }
    Employee6(int eno,String ename,double sal) // Parameterised const, eno1,ename1..-local variables
    {
        super(eno);// super keyword always calls super class constructor.
        this.ename=ename;
        this.sal=sal;
    }
    public void display()//method
    {
        System.out.println("Employee number:"+eno);
        System.out.println("Employee name:"+ename);
        System.out.println("Employee salary:"+sal);
    }
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        new Employee6();// Object Creation
        Employee6 emp6=new Employee6(21,"Dhruti",600000);//object reference
        emp6.display();
    }
```

Problems | Javadoc | Declaration | Console ×

```
<terminated> Employee6 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-
Default Constructor
Employee No super class:21
Employee number:0
Employee name:Dhruti
Employee salary:600000.0
```

Employee2.java | Employee3.java | Employee4.java | Employee

```java
package com.evergent.corejava.constructor;
class Animal
{
    private String name;
    private int age;
    //constructor
    public Animal(String name, int age)
    {
        this.name=name;
        this.age=age;
    }
    //method
    public void displayInfo()
    {
        System.out.println("name:"+name);
        System.out.println("age:"+age);
    }
}
//subclass inheritance
class Dog extends Animal
{
    private String breed;
    public Dog(String name,int age, String breed)
    {
        super(name,age);//call to super class constructor
        this.breed=breed;
    }
    //method overriding
    public void displayInfo()
    {
        super.displayInfo();
        System.out.println("Breed:"+breed);
    }
}
public class Inheritance_Overriding8 {
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Dog dog=new Dog("Buddy",6,"Golden Retriever");
        dog.displayInfo();
    }
```

Problems | Javadoc | Declaration | Console ×

```
<terminated> Inheritance_Overriding8 [Java Application] C:\Users\dhruti.guthik
age:6
Breed:Golden Retriever
```

Employee2.java | Employee3.java | Employee4.java | E

```java
package com.evergent.corejava.constructor;
class Car
{
    String color;
    int maxspeed;
    //default constructor
    Car()
    {
        color="white";
        maxspeed=120;
    }
    //parameterised constructor
    Car(String color,int maxspeed)
    {
        this.color=color;
        this.maxspeed=maxspeed;
    }
    void display()
    {
        System.out.println("color:"+color);
        System.out.println("maxspeed:"+maxspeed);
    }
}

public class MyCars7 {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Car car1=new Car();
        Car car2=new Car("Red",150);
        car1.display();
        car2.display();
    }
}

```

Problems | Javadoc | Declaration | Console ×

```
<terminated> MyCars7 [Java Application] C:\Users\dhruti.guthikonda\
color:white
maxspeed:120
color:Red
maxspeed:150
```

Employee3.java | Employee4.java | Employee5.java | Employe

```
1  package com.evergent.corejava.constructor;
2
3  public class Student9 {
4      String name;
5      int age;
6      //constructor
7      public Student9(String name,int age)
8      {
9          this.name=name;
10         this.age=age;
11     }
12     //copy Constructor
13     public Student9(Student9 s)
14     {
15         this.name=s.name;
16         this.age=s.age;
17     }
18     //method
19     public void displayDetails()
20     {
21         System.out.println("Name:"+name);
22         System.out.println("age:"+age);
23     }
24
25     public static void main(String[] args) {
26         // TODO Auto-generated method stub
27         Student9 student1= new Student9("dhruti",21);
28         Student9 student2=new Student9(student1);
29         student1.displayDetails();
30         student2.displayDetails();
31
32     }
33
34 }
35
```

Problems | Javadoc | Declaration | Console

```
<terminated> Student9 [Java Application] C:\Users\dhruti.guthikonda\Deskto
Name:dhruti
age:21
Name:dhruti
age:21
```

Day5:(9-08-24)

1. Static

Employee4.java | Employee5.java | Employee6.java | MyCars7.java

```
1  package com.evergent.corejava.static1;
2  //static variable and method declaration
3  public class StaticDemo1 {
4      static String cname="India";//static variables or class vari
5      static public void myData()
6      {
7          System.out.println("MyData");
8      }
9
10     public static void main(String[] args) {
11         // TODO Auto-generated method stub
12         System.out.println(StaticDemo1.cname);
13         StaticDemo1.myData();
14
15     }
16
17 }
18
```

Problems | Javadoc | Declaration | Console

```
<terminated> StaticDemo1 [Java Application] C:\Users\dhruti.guthikonda\Desktop\ecl
India
MyData
```

Employee5.java | Employee6.java | MyCars7.java | Inheritance... | Student9.java

```
1  package com.evergent.corejava.static1;
2  //Static methods can access static methods and static variables only
3  public class StaticDemo2 {
4      static String cname="India";
5      String name="Dhruti";//static variables or class variables
6      static public void myData()
7      {
8          System.out.println("MyData");
9      }
10     public void show()
11     {
12         System.out.println("Show is non static method");
13     }
14
15     public static void main(String[] args) {
16         // TODO Auto-generated method stub
17         myData();
18         //show();
19         //Cannot make a static reference to the non-static method show() f
20         System.out.println(cname);
21
22     }
23 }
24
```

Problems | Javadoc | Declaration | Console

```
<terminated> StaticDemo2 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03
MyData
India
```

Employee6.java | MyCars7.java | Inheritance... | Student9.java | CustomerSe

```
1  package com.evergent.co
2  //Static methods can ac
3  public class StaticDemo3 {
4      static String cname="India";
5      String name="Dhruti";//static variables or class variables
6      static public void myData()
7      {
8          //Error: Cannot make a static reference of non static method()
9          //show();
10         System.out.println("MyData");
11     }
12     public void show()
13     {
14         System.out.println("Show is non static method");
15     }
16
17     public static void main(String[] args) {
18         // TODO Auto-generated method stub
19         myData();
20         //show();
21         //Cannot make a static reference to the non-static method show(
22         System.out.println(cname);
23
24     }
25
26 }
27
```

coreJAVA_Development/src/com/evergent/corejava/constru

Problems | Javadoc | Declaration | Console

```
<terminated> StaticDemo3 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023
MyData
India
```

MyCars7.java | Inheritance... | Student9.java | CustomerServ... | StaticDemo1....

```
1  package com.evergent.corejava.static1;
2  //Non static method can access static variables and static method
3  public class StaticDemo4 {
4      static String cname="India";
5      String name="Dhruti";//static variables or class variables
6      static public void myData()
7      {
8          //Error: Cannot make a static reference of non static method()
9          //show();
10         System.out.println("MyData");
11     }
12     public void show()
13     {
14         myData();
15         System.out.println("Show is non static method"+cname);
16     }
17
18     public static void main(String[] args) {
19         // TODO Auto-generated method stub
20         myData();
21         //show();
22         //Cannot make a static reference to the non-static method show() f
23         System.out.println(cname);
24         StaticDemo4 sd4=new StaticDemo4();
25         sd4.show();
26     }
27 }
```

Problems | Javadoc | Declaration | Console

```
<terminated> StaticDemo4 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03
MyData
India
MyData
Show is non static methodIndia
```

```java
package com.evergent.corejava.static1;
//static variable and method declaration
public class StaticDemo5 {
    static String cname="India";//static variables or class variables
    static
    {
        System.out.println("static Block");
    }
    static public void myData()
    {
        System.out.println("MyData");
    }

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        System.out.println(StaticDemo5.cname);
        StaticDemo5.myData();

    }

}
```

```
<terminated> StaticDemo5 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-20
static Block
India
MyData
```

```java
package com.evergent.corejava.static1;

public class Person6 {
    static String name="Dhruti";
    int age=22;
    String address="Hyd";
    public void display()
    {
        name="Welcome";
        System.out.println("name:"+name);
        System.out.println("age:"+age);
        System.out.println("address:"+address);
    }

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Person6 p1= new Person6();
        System.out.println(p1.name);
        Person6 p2=new Person6();
        System.out.println(p2.name);

    }

}
```

```
<terminated> StaticDemo5 [Java Application] C:\Users\dhruti.guthikor
static Block
India
MyData
```

2. Final

```java
package com.evergent.corejava.final1;

public class FinalDemo1 {
    final String cname="India";
    public void myData()
    {
        //cname="Welcome"; gives error
        System.out.println("cname:"+cname);
    }

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        FinalDemo1 fd= new FinalDemo1();
        fd.myData();

    }

}
```

```
<terminated> FinalDemo1 [Java Application] C:\Users\dhruti.guthik
cname:India
```

```java
package com.evergent.corejava.final1;
//we cant override final method and inheit
class MyClass
{
    final public void myProducts()
    {
        System.out.println("All products ");
    }
}

public class FinalDemo2 extends MyClass{
    final String cname="India";
    //cannot overide the final method from MyClass
    public void myProducts1()
    {
        System.out.println("Hello Products");
    }
    public void myData()
    {
        System.out.println("cname:"+cname);
    }

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        FinalDemo2 fd =new FinalDemo2();
        fd.myData();
        fd.myProducts();
    }
}
```

```
<terminated> FinalDemo1 [Java Application] C:\Users\dhruti.guthikonda\D
cname:India
```

```java
package com.evergent.corejava.final1;
//we cant override final method and inherit
final class MyClass1
{
    final public void myProducts()
    {
        System.out.println("All products");
    }
}
//The type FinalDemo3 cannot subclass the final class MyClass
public class FinalDemo3 {//extends MyClass
    final String cname="India";
    //cannot overide the final method from MyClass
    public void myProducts1()
    {
        System.out.println("Hello Products");
    }
    public void myData()
    {
        System.out.println("cname:"+cname);
    }
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        FinalDemo3 fd =new FinalDemo3();
        MyClass1 mc=new MyClass1();
        fd.myData();
        mc.myProducts();
    }
}
```

```
<terminated> FinalDemo3 [Java Application] C:\Users\dhruti.guthikonda\Desktop\ecl
cname:India
All products
```

Day6:(12-08-24)

1. Strings

StaticDemo4.... StaticDemo5.... Person6.java FinalDemo

```
1  package com.evergent.corejava.strings;
2
3  public class StringDemo1 {
4
5      public static void main(String[] args) {
6          // TODO Auto-generated method stub
7          String str1=new String("JAVA");
8          String str2=new String("JAVA");
9          if (str1==str2) // checks for memory location
10             System.out.println("True");
11         else
12             System.out.println("False");
13         if(str1.equals(str2)) // checks for content
14             System.out.println("True");
15         else
16             System.out.println("False");
17
18
19     }
20
21 }
22
```

Problems Javadoc Declaration Console ×

```
<terminated> StringDemo1 [Java Application] C:\Users\dhruti.guthikonda\
False
True
```

StaticDemo5.... Person6.java FinalDemo1.java FinalDemo2.java FinalDemo3.jav

```
1  package com.evergent.corejava.strings;
2  // string constant pool concept
3  // here there is no use of new keyword so it directly checks in constant pool
4  public class StringDemo2 {
5
6      public static void main(String[] args) {
7          // TODO Auto-generated method stub
8          String s1= "Java";
9          String s2="Java";
10         if (s1==s2) // checks for memory location
11             System.out.println("True");
12         else
13             System.out.println("False");
14         if(s1.equals(s2)) // checks for content
15             System.out.println("True");
16         else
17             System.out.println("False");
18
19
20     }
21
22 }
```

Problems Javadoc Declaration Console ×

```
<terminated> StringDemo2 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\ecli
True
True
```

Person6.java FinalDemo1.java FinalDemo2.java FinalDemo3

```
1  package com.evergent.corejava.strings;
2
3  public class StringDemo3_Methods {
4
5      public static void main(String[] args) {
6          // TODO Auto-generated method stub
7          String name= new String("    Hello Dhruti");
8          System.out.println(name.length());
9          System.out.println(name.toLowerCase());
10         System.out.println(name.toUpperCase());
11         System.out.println(name.trim());
12
13
14     }
15
16 }
17
```

Problems Javadoc Declaration Console ×

```
<terminated> StringDemo3_Methods [Java Application] C:\Users\dhruti.guthik
16
    hello dhruti
    HELLO DHRUTI
Hello Dhruti
```

STRINGS

1. String Class
2. String Buffer
3. String Builder

I. String Class

a) String class is final

b) String is immutable

c) All string class methods are non synchronized

II. String Buffer

a) String Buffer class is final

b) String Buffer is mutable

c) All string buffer class methods are synchronized

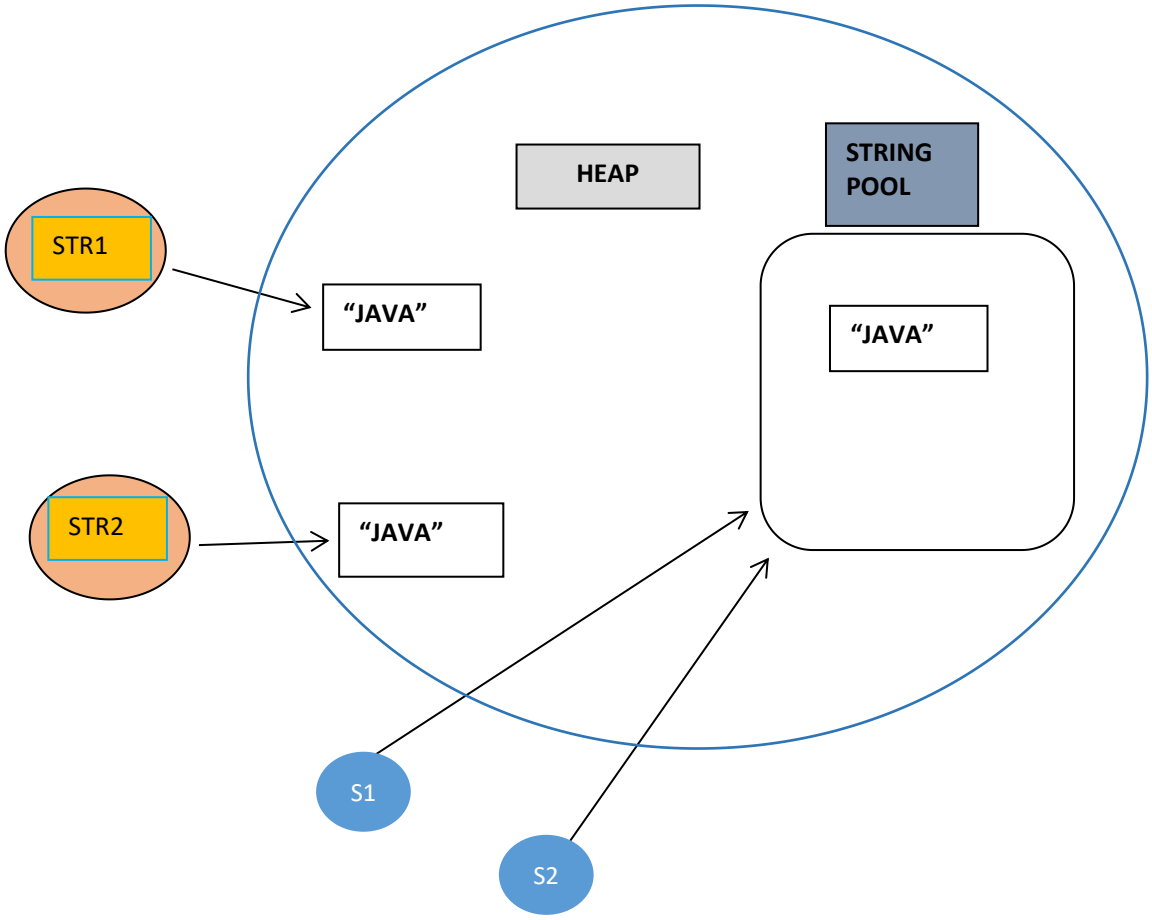
III. String Builder

a) String Builder class is final

b) String Builder is mutable

c) All string builder class methods are non synchronized

S.No	**String**	**String Buffer**	**String Builder**
1.	Final class	Final class	Final class
2.	Immutable	Mutable	Mutable
3.	Non Synchronized(Not Thread safe)	Synchronized(Thread safe)	Non Synchronized(Not Thread safe)
4.	String constant pool area	Heap area	Heap area



String class points

i. In java a string is a sequence of characters, often used to represent text.

ii. String are objects in java & instance of string class which is part of java.lang.package.

iii. Key features of string in java

   a) Immutable : Once a string object is created it cannot be changed.

   b) Any modification to string creates a new string object

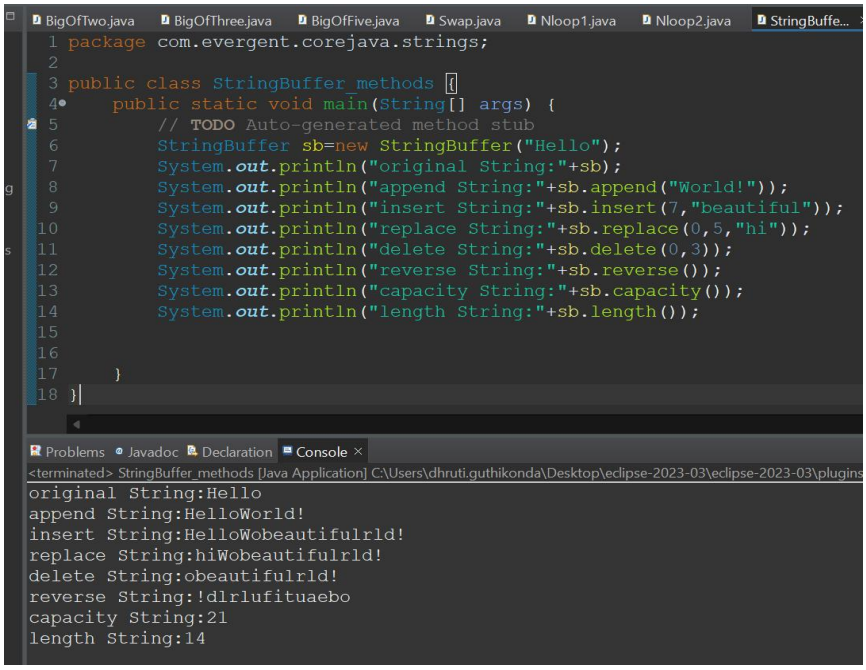
iv. Java optimizes memory usage by storing strings in a special way area of memory called "String Constant Pool".

v. If two strings have same value and are created without using new keyword they will refer to same object in string pool.

vi. We can create string using in two ways :

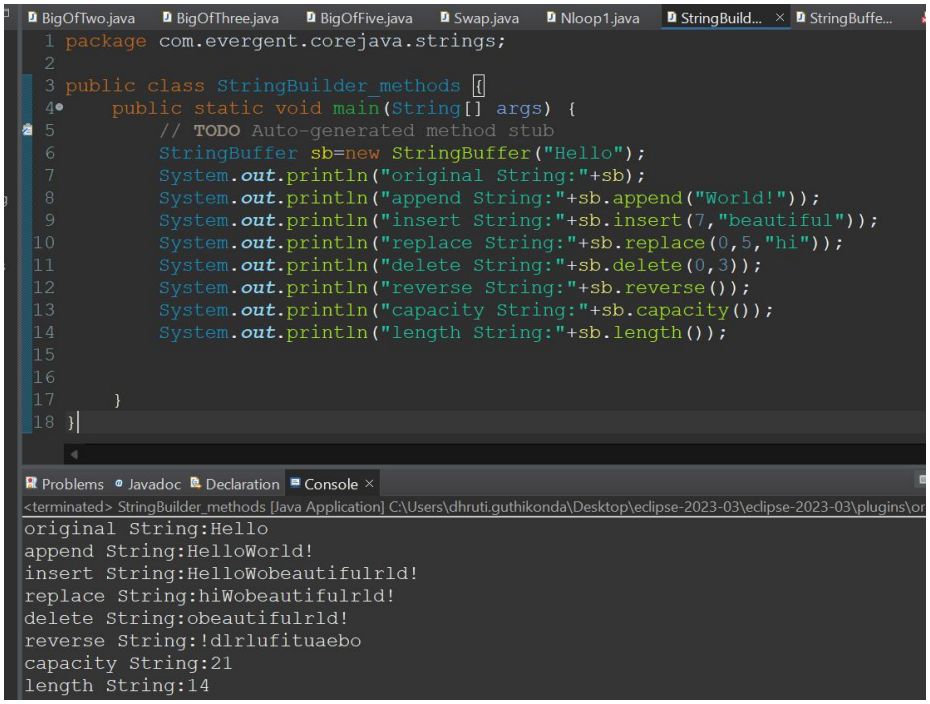
   a) Using String literals

   b) Using new keyword



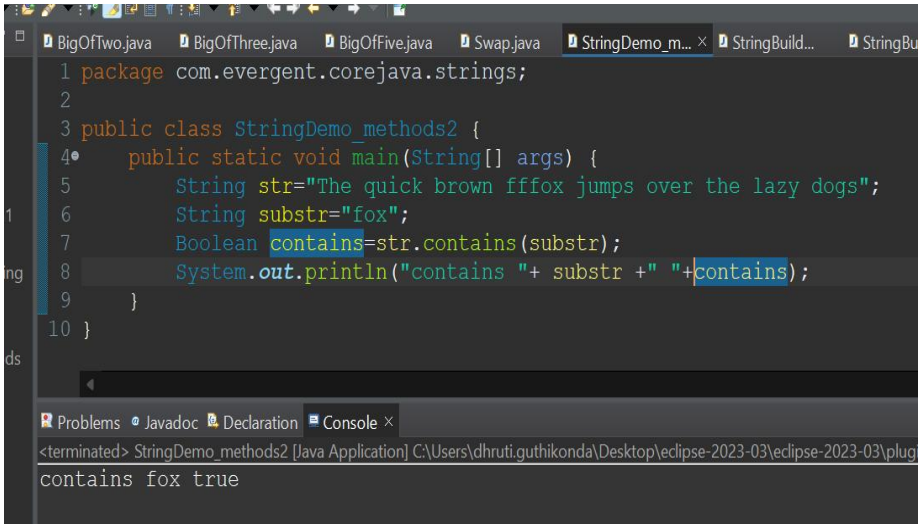
```java
package com.evergent.corejava.strings;

public class StringBuffer_methods {
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        StringBuffer sb=new StringBuffer("Hello");
        System.out.println("original String:"+sb);
        System.out.println("append String:"+sb.append("World!"));
        System.out.println("insert String:"+sb.insert(7,"beautiful"));
        System.out.println("replace String:"+sb.replace(0,5,"hi"));
        System.out.println("delete String:"+sb.delete(0,3));
        System.out.println("reverse String:"+sb.reverse());
        System.out.println("capacity String:"+sb.capacity());
        System.out.println("length String:"+sb.length());


    }
}
```

```
original String:Hello
append String:HelloWorld!
insert String:HelloWobeautifulrld!
replace String:hiWobeautifulrld!
delete String:obeautifulrld!
reverse String:!dlrlufituaebo
capacity String:21
length String:14
```

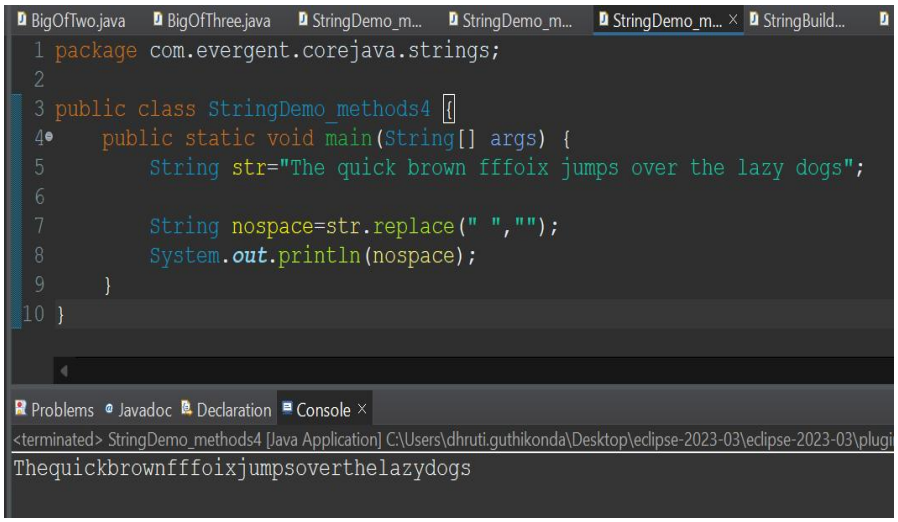


```java
package com.evergent.corejava.strings;

public class StringBuilder_methods {
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        StringBuffer sb=new StringBuffer("Hello");
        System.out.println("original String:"+sb);
        System.out.println("append String:"+sb.append("World!"));
        System.out.println("insert String:"+sb.insert(7,"beautiful"));
        System.out.println("replace String:"+sb.replace(0,5,"hi"));
        System.out.println("delete String:"+sb.delete(0,3));
        System.out.println("reverse String:"+sb.reverse());
        System.out.println("capacity String:"+sb.capacity());
        System.out.println("length String:"+sb.length());


    }
}
```

```
original String:Hello
append String:HelloWorld!
insert String:HelloWobeautifulrld!
replace String:hiWobeautifulrld!
delete String:obeautifulrld!
reverse String:!dlrlufituaebo
capacity String:21
length String:14
```


BigOfTwo.java
BigOfThree.java
BigOfFive.java
Swap.java
StringDemo_m...
StringBuild...
1 package com.evergent.corejava.strings;
2
3 public class StringDemo_methods2 {
4 public static void main(String[] args) {
5 String str="The quick brown fffox jumps over the lazy dogs";
6 String substr="fox";
7 Boolean contains=str.contains(substr);
8 System.out.println("contains "+ substr +" "+contains);
9 }
10 }
Problems
Javadoc
Declaration
Console
<terminated> StringDemo_methods2 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023-03\plug
contains fox true


1 package com.evergent.corejava.strings;
2
3 public class StringDemo_methods3 {
4 public static void main(String[] args) {
5 String str="The quick brown fffoix jumps over the lazy dogs";
6 String substr="fox";
7 Boolean contains=str.contains(substr);
8 System.out.println("contains "+substr+" "+contains);
9 }
10 }
Problems
Javadoc
Declaration
Console
<terminated> StringDemo_methods3 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023-03\plug
contains fox false


BigOfTwo.java
BigOfThree.java
StringDemo_m...
StringDemo_m...
StringDemo_m...
StringBuild...
1 package com.evergent.corejava.strings;
2
3 public class StringDemo_methods4 {
4 public static void main(String[] args) {
5 String str="The quick brown fffoix jumps over the lazy dogs";
6
7 String nospace=str.replace(" ","");
8 System.out.println(nospace);
9 }
10 }
Problems
Javadoc
Declaration
Console
<terminated> StringDemo_methods4 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023-03\plugi
Thequickbrownfffoixjumpsoverthelazydogs

```java
package com.evergent.corejava.strings;

public class StringDemo_methods5 {
    public static void main(String[] args) {
        String str="The quick brown fffoix jumps over the lazy dogs";

        StringBuilder reversed=new StringBuilder(str).reverse();
        System.out.println(reversed);
    }
}
```

Problems Javadoc Declaration Console

<terminated> StringDemo_methods5 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023-03\plu

```
sgod yzal eht revo spmuj xiofff nworb kciuq ehT
```

```java
package com.evergent.corejava.strings;

public class SplitDemo1 {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        String str="Java is a powerful language";
        //Split the string by space
        String[] words= str.split(" ");
        for (int i=0;i<words.length;i++)
        System.out.println(words[i]);
        //for each adv loop
        for(String w:words)
        {
            System.out.println(w);
        }
    }
}
```

Problems Javadoc Declaration Console

<terminated> SplitDemo1 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-0

```
Java
is
a
powerful
language
Java
is
a
powerful
language
```

```java
package com.evergent.corejava.strings;

public class SplitDemo2 {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        String str="Java is a powerful language";
        //Split the string by space
        String[] words= str.split(" ");
        for (int i=0;i<words.length;i++)
        System.out.println(words[i]);


    }

}
```

Problems Javadoc Declaration Console

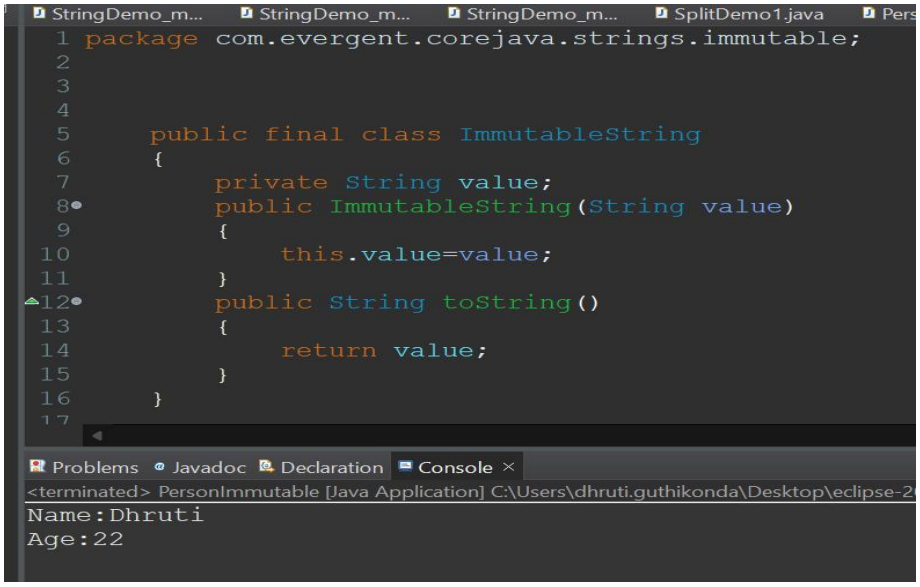
```
<terminated> SplitDemo2 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\e
Java
is
a
powerful
language
```

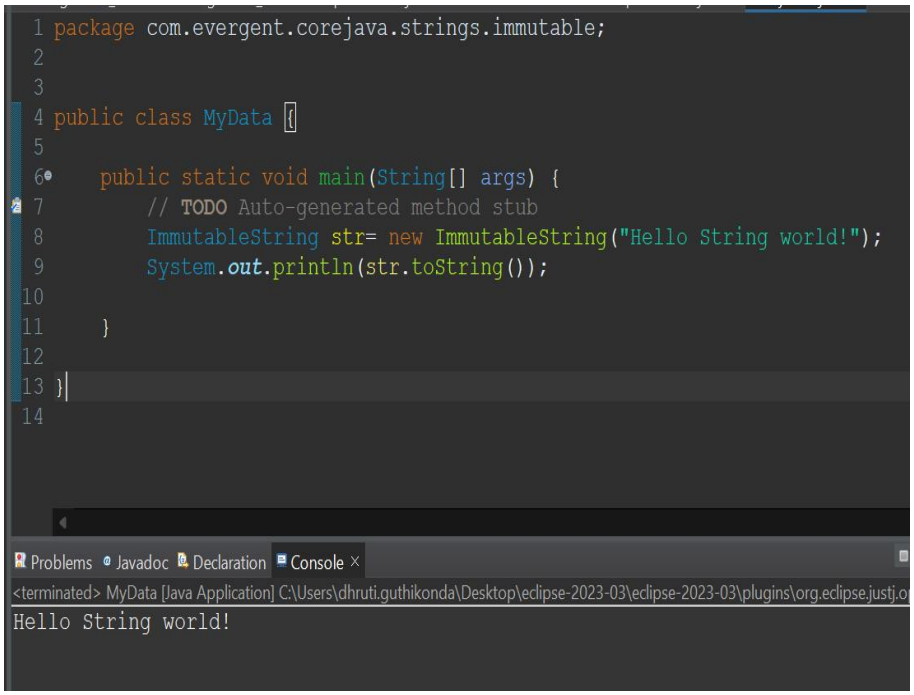
StringDemo_m... StringDemo_m... StringDemo_m... StringDemo_m... SplitDemo1.java Person

```java
package com.evergent.corejava.strings.immutable;
//Q- Making String Immutable
public final class PersonImmutable {
    private final String name;
    private final int age;
    //Constructor to initialize the fields
    public PersonImmutable(String name, int age)
    {
        this.name=name;
        this.age=age;
    }
    public String MyName()
    {
        return name;
    }
    public int MyAge()
    {
        return age;
    }
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        PersonImmutable person =new PersonImmutable("Dhruti",22);
        System.out.println("Name:"+ person.MyName());
        System.out.println("Age:"+ person.MyAge());
    }
}
```

Problems Javadoc Declaration Console

```
<terminated> PersonImmutable [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023-03\p
Name:Dhruti
Age:22
```


StringDemo_m...
StringDemo_m...
StringDemo_m...
SplitDemo1.java
1 package com.evergent.corejava.strings.immutable;
2
3
4
5 public final class ImmutableString
6 {
7 private String value;
8 public ImmutableString(String value)
9 {
10 this.value=value;
11 }
12 public String toString()
13 {
14 return value;
15 }
16 }
Problems
Javadoc
Declaration
Console
<terminated> PersonImmutable [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2
Name:Dhruti
Age:22


1 package com.evergent.corejava.strings.immutable;
2
3
4 public class MyData {
5
6 public static void main(String[] args) {
7 // TODO Auto-generated method stub
8 ImmutableString str= new ImmutableString("Hello String world!");
9 System.out.println(str.toString());
10
11 }
12
13 }
14
Problems
Javadoc
Declaration
Console
<terminated> MyData [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023-03\plugins\org.eclipse.justj.o
Hello String world!

Day7:(13-08-24)
INTERFACES:

1. Interface is a keyword.
2. We can declare methods signature but not implementation
3. By default all interface methods are abstract.
4. If any class implements interface, that class should be override all interface methods, other wise that class will be showing compile time error.
5. We cannot create object to interface but we can create reference to interface.
6. We can declare variables inside interface and all interface variables are static final.
7. Java will support multiple inheritance through interface.
8. One class can implement interfaces.
9. One interface can extend other interface.
10. We can declare interfaces without methods.(0 Methods)
11. Clonable, serialaxable inerface are mechanisms and are called marker interfaces.

BigOfTwo.java | BigOfThree.java | BigOfFive.java | MyData.java | Book.ja

```java
package com.evergent.corejava.interfaces;

public interface Book {
    String cname="India";
    public void bookTitle();
    public void bookAuthor();
    public void bookPrice();

}

```

BigOfTwo.java | BigOfThree.java | MyData.java | Book.java | BookIm

```java
package com.evergent.corejava.interfaces;

public class BookImpl implements Book {
    public void bookTitle()
    {
        System.out.println("Corejava: "+cname);
    }
    public void bookAuthor()
    {
        System.out.println("Oracle crop");
    }
    public void bookPrice()
    {
        System.out.println("price is: 550");
    }
    public void show()
    {
        System.out.println("Show is a local method");
    }
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        BookImpl book1= new BookImpl();
        book1.bookTitle();
        book1.bookAuthor();
        book1.bookPrice();
        book1.show();
    }
}
```

Problems | Javadoc | Declaration | Console ×

<terminated> BookImpl [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-

```
Corejava: India
Oracle crop
price is: 550
Show is a local method
```

```
        System.out.println("Corejava: "+cname);
    }
    public void bookAuthor()
    {
        System.out.println("Oracle crop");
    }
    public void bookPrice()
    {
        System.out.println("price is: 550");
    }
    public void show()
    {
        System.out.println("Show is a local method");
    }

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        //can not instantiate the type Book
        //Book b1= newBook();
        Book b2= new BookImpl2();// creating reference for interface but not class
        b2.bookTitle();
        b2.bookAuthor();
        b2.bookPrice();
        //b2.show();
        //The method show() is undefined for the type Book
    }
}

```

Problems Javadoc Declaration Console ×

```
<terminated> BookImpl2 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023-03\plugins\org.eclipse.j
Corejava: India
Oracle crop
price is: 550
```

BigOfTwo.java MyData.java Book.java BookImpl.java BookImpl2.j

```
        {
            System.out.println("price is: 550");
        }
        public void show()
        {
            System.out.println("Show is a local method");
        }
        public void addNewBook()
        {
            System.out.println("Java New Version");
        }
        public void myData()
        {
            System.out.println("Mydata interface");
        }

        public static void main(String[] args) {
            // TODO Auto-generated method stub
            BookImpl3 book1= new BookImpl3();
            book1.bookTitle();
            book1.bookAuthor();
            book1.bookPrice();
            book1.show();
            book1.addNewBook();
            book1.myData();
        }
}

```

Problems Javadoc Declaration Console ×

```
<terminated> BookImpl3 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2
Corejava: India
Oracle crop
price is: 550
Show is a local method
Java New Version
Mydata interface
```

MyData.java Book.java BookImpl.java BookImpl2.java Boo

```java
package com.evergent.corejava.interfaces;

public interface MyDataInfo {
    public void myData();

}

```

Book.java BookImpl.java BookImpl2.java BookImpl3.java

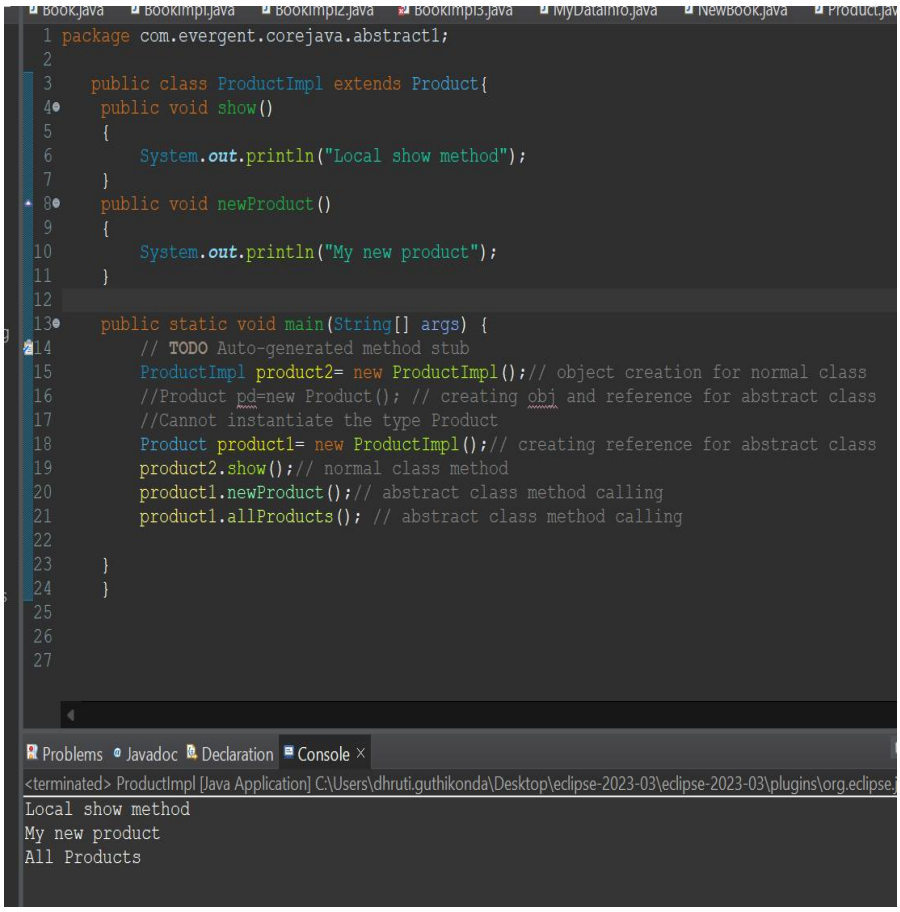
```java
package com.evergent.corejava.interfaces;

public interface NewBook extends MyDataInfo{
    public void addnewBook();
    public void bookPrice();

}

```

Day8:(14-08-24)
ABSTRACT CLASS:

1. Abstract is a keyword.
2. Abstract class having abstract methods and concrete(implemented) methods.
3. If any class having one abstract method that class should be declared as a Abstract keyword otherwise that class will be showing a compile time error.
4. If any class extends abstract class , that class should be override all abstract class methods otherwise , the class will be showing compile time error.
5. We cannot create object to abstract class but we can create reference to abstract class.

Book.java BookImpl.java BookImpl2.java BookImpl3.java MyDataInfo.java

```java
package com.evergent.corejava.abstract1;

 abstract public class Product {
     abstract public void newProduct(); // Abstract methods
    public void allProducts()// Normal local method
    {
        System.out.println("All Products");
    }
}

```



Day10:(19-08-24)
EXCEPTION HANDLING:

1. Exception handling is a mechanism.
2. Exception handling is a inbuilt mechanism.
3. All exceptions are executed while abnormal conditions only.
4. If there is normal flow then it wont execute any exceptions.
5. Once any exception occur in the java code then remaining lines of code is unreachable.
6. Java.lang.Throwable is super class for exceptions and errors.
7. There are two types of exceptions in java:
    a. Checked Exception.
    b. Unchecked Exception.
8. All Checked exceptions are compile time exceptions.
9. All Unchecked exceptions are runtime exceptions.
10. There are 5 keywords in Exception Handling:
    a. Try
    b. Catch
    c. Finally
    d. Throw
    e. Throws
11. 'Try' is used for business logic
12. 'catch' is used for handling the exception.
13. 'finally' is a block, which is executed whether exception occur or not nad also used to close database connections.
14. Throws an exception will be executed method by method.
15. Throw is used for runtime exceptions and will call predefined exceptions.
16. Try must be followed by either catch block or finally block.
17. We should follow exceptions heirarchical.
18. We can crate our own exceptions called userdefined exceptions.
19. Our own exceptions extends Exceptions or Runtime Exceptions.
20. All Exception classes are in to java.lang package.
21. If there are two exceptions in a class, developer should handle 1st exception and then 2nd exception will be handled.
22. Errors are not in our control. Errors cant be handled.
23. Throw is a keyword will call both compile time and runtime errors.
24. Throws work on compile time.

<> ArrayIndexOutOfBoundException

```java
package com.evergent.corejava.exceptionhandling;

public class ArrayIndexOutOfException13 {

    public static void main(String[] args) {
        int[] numbers= {1,2,3,4,5};
        try {
            System.out.println("Accesing element at index 10: "+numbers[10]);
        }
        catch(ArrayIndexOutOfBoundsException e){
            System.out.println("Caught an exception "+e.getMessage());
        }
        System.out.println("Program continues after exception handling");
    }
}

```

```
Console ×
<terminated> ArrayIndexOutOfException13 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023-03\plugins\org.ecl
Caught an exception Index 10 out of bounds for length 5
Program continues after exception handling
```

<> CommandLineArgument

```java
package com.evergent.corejava.exceptionhandling;

public class CommandLineArgument14 {
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        System.out.println(args[0]);
        System.out.println(args[1]);
    }
}


```

```
Console ×
<terminated> CommandLineArgument14 [Java Application] C:\Users\dhruti.guthikonda\Desktop
100
200
```

<> CompileTime

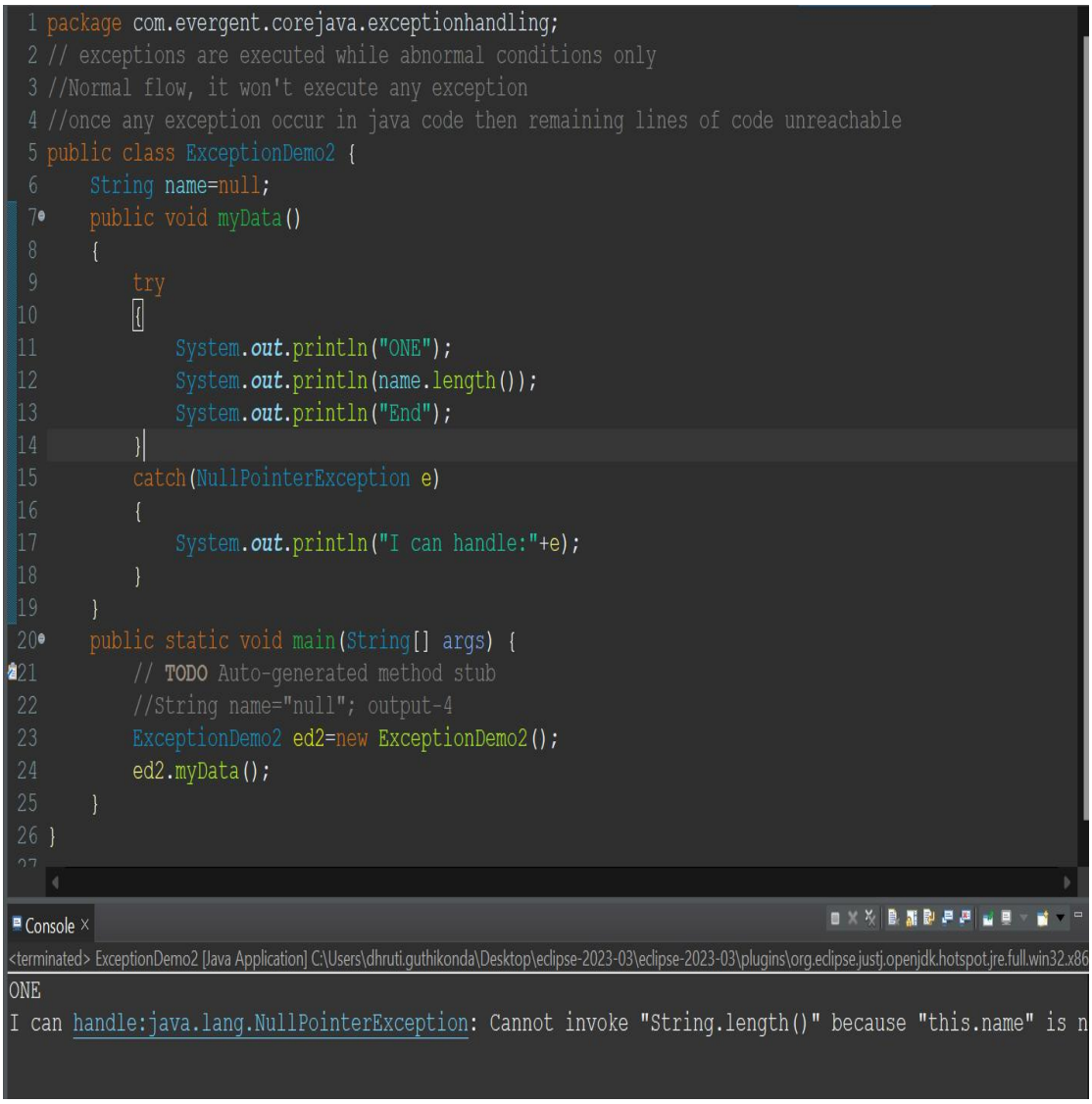
```java
package com.evergent.corejava.exceptionhandling;
import java.io.File;
public class CompileTimeDemo15 {
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        try {
            File file=new File("C:\\myInfo\\myData");
            Scanner scanner=new Scanner(file);
            while(scanner.hasNextLine()) {
                System.out.println(scanner.nextLine());
            }
            scanner.close();
        }
        catch(FileNotFoundException e){
            e.printStackTrace();
        }
    }
}
```

<> Inbuilt Mechanism

```java
package com.evergent.corejava.exceptionhandling;
//1. Exceptions are inbuilt mechanism in java
public class ExceptionDemo1 {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        //String name="null"; output-4
        String name=null;
        System.out.println(name.length());
    }
}
```

```
Console ×
<terminated> ExceptionDemo1 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023-03\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_
Exception in thread "main" java.lang.NullPointerException: Cannot invoke "String.length()" because "n
        at com.evergent.corejava.exceptionhandling.ExceptionDemo1.main(ExceptionDemo1.java:9)
```

<>exceptions are executed while abnormal conditions only
Normal flow, it won't execute any exception
once any exception occur in java code then remaining lines of code unreachable



```java
package com.evergent.corejava.exceptionhandling;
// exceptions are executed while abnormal conditions only
//Normal flow, it won't execute any exception
//once any exception occur in java code then remaining lines of code unreachable
public class ExceptionDemo2 {
    String name=null;
    public void myData()
    {
        try
        {
            System.out.println("ONE");
            System.out.println(name.length());
            System.out.println("End");
        }
        catch(NullPointerException e)
        {
            System.out.println("I can handle:"+e);
        }
    }
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        //String name="null"; output-4
        ExceptionDemo2 ed2=new ExceptionDemo2();
        ed2.myData();
    }
}
```

```
Console ×
<terminated> ExceptionDemo2 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023-03\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86
ONE
I can handle:java.lang.NullPointerException: Cannot invoke "String.length()" because "this.name" is n
```

<> Multiple Exceptions

```java
package com.evergent.corejava.exceptionhandling;
// Multiple exception
import java.lang.*;
class ArithmeticException extends Throwable
{
}
public class ExceptionDemo3 {
    String name = null;
    int k = 2;

    public void myData() {
        try {
            System.out.println("one");
            System.out.println(name.length());
            int t = 10 / k;
            System.out.println("End");
        } catch (NullPointerException e) {
            System.out.println("I can handle: " + e);
        } catch (ArithmeticException e) {
            System.out.println("I can handle: " + e);
        }
    }
            /*catch(Exception e)
            {
                System.out.println("I can handle:"+e);
            }*/
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        ExceptionDemo3 ed3=new ExceptionDemo3();
        ed3.myData();
    }
}
```

<> Finally block

```java
package com.evergent.corejava.exceptionhandling;
//exception occur or not finally block will get excuted
public class ExceptionDemo5finally {
    String name=null;
    int k=0;
    public void myData()
    {
        try
        {
            System.out.println("one");
            System.out.println(name.length());
            int t=10/k;
            System.out.println("End");
        }
            catch(NullPointerException e)
            {
                System.out.println("I can handle:"+e);
            }
            //catch(ArithmeticException e)
            //{
                //System.out.println("I can handle:"+e);
            //}
        catch(Exception e)
        {
            System.out.println(e);
        }
        finally
        {
            System.out.println("Finally block for closing db or network connections");
        }
    }
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        ExceptionDemo5finally ed3=new ExceptionDemo5finally();
        ed3.myData();
    }
}
```

Console

```
<terminated> ExceptionDemo5finally [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023-03\plugins\org.eclipse.justj.openjdk.hotsp
one
I can handle:java.lang.NullPointerException: Cannot invoke "String.length()" because "this.name" is null
Finally block for closing db or network connections
```

<> try should be followed by catch or finally

```java
package com.evergent.corejava.exceptionhandling;
// try should be followed by either catch or finally block
public class ExceptionDemo6finally2 {
    String name="null";
    int k=2;
    public void myData()
    {
        try
        {
            System.out.println("one");
            System.out.println(name.length());
            int t=10/k;
            System.out.println("End");
        }
            /*catch(NullPointerException e)
            {
                System.out.println("I can handle:"+e);
            }
            catch(ArithmeticException e)
            {
                System.out.println("I can handle:"+e);
            }*/
        finally
        {
            System.out.println("Finally block for closing db or network connections");
        }
    }
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        ExceptionDemo6finally2 ed3=new ExceptionDemo6finally2();
        ed3.myData();
    }
}
```

Console

```
<terminated> ExceptionDemo6finally2 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-
one
4
End
Finally block for closing db or network connections
```

<> Exceptions are executed while abnormal conditions
Normal flow, it wont execute any exception
Once any exception occur in java code then remaining lines of code ar
Unreachable

```java
package com.evergent.corejava.exceptionhandling;
// exceptions are executed while abnormal conditions only
//Normal flow, it won't execute any exception
//once any exception occur in java code then remaining lines of code unreachable
public class ExceptionDemo7 {
    String name=null;
    public void myData() throws NullPointerException
    {
        System.out.println("ONE");
        System.out.println(name.length());
        System.out.println("End");


    }
    public static void main(String[] args) {
        try
        {
            ExceptionDemo7 ed2=new ExceptionDemo7();
            ed2.myData();
        }
        // TODO Auto-generated method stub
        //String name="null"; output-4
        catch(NullPointerException e)
        {
            System.out.println("I can handle:"+e);
        }
    }
}
```

Console X

<terminated> ExceptionDemo7 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023-03\plugins\org.eclipse.justj.openjdk.hotspot.jre.f

```
ONE
I can handle:java.lang.NullPointerException: Cannot invoke "String.length()" because "this.name" is null
```

<> Heap error

```java
package com.evergent.corejava.exceptionhandling;
//Heap error
public class MyoutofMemory17 {

    public static void main(String[] args) throws Exception {
        // TODO Auto-generated method stub
        Integer[] array=new Integer[100000*1000000];
        System.out.println(array);
    }

}
```

Console ×

<terminated> MyoutofMemory17 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023

```
[Ljava.lang.Integer;@515f550a
```

<> ProductNotFoundException

```java
package com.evergent.corejava.exceptionhandling;
class ProductNotFoundException extends Exception
{
    public ProductNotFoundException(String message) {
        System.out.println("Hello:"+message);
    }
}
public class ProductImpl {
    int pno=105;
    public void myData() throws ProductNotFoundException
    {
        if(pno>100)
            throw new ProductNotFoundException("this product is not available");
        else
            System.out.println("Product is there..");
    }
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        try
        {
            ProductImpl product1= new ProductImpl();
            product1.myData();
        }
        catch(Exception e) {
            System.out.println("I can handle:"+e);
        }
    }
}
```

Console ×

<terminated> ProductImpl (2) [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023-03\plugins\or

```
Hello:this product is not available
I can handle:com.evergent.corejava.exceptionhandling.ProductNotFoundException
```

<> Age Not support

```java
package com.evergent.corejava.exceptionhandling;
class AgeNotSupport extends Exception
{
    public AgeNotSupport(String message) {
        System.out.println("Hello:"+message);
    }
}
public class ProductImpl2 {
    int age=25;
    public void myData() throws AgeNotSupport
    {
        if(age>18)
            throw new AgeNotSupport("Eligible for voting");
        else
            System.out.println("not eligible..");
    }
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        try
        {
            ProductImpl2 product1= new ProductImpl2();
            product1.myData();
        }
        catch(Exception e) {
            System.out.println("I can handle:"+e);
        }
    }
}
```

```
Console ×
<terminated> ProductImpl2 (1) [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023-
Hello:Eligible for voting
I can handle:com.evergent.corejava.exceptionhandling.AgeNotSupport
```

<> Stackoverflow error

```java
package com.evergent.corejava.exceptionhandling;

public class StackOverflowErrorExample {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        try
        {
            recursiveMethod();
        }
        catch(StackOverflowError e)
        {
            System.out.println("StackOverflow errror caught:"+e.getMessage());
        }
    }
    //recursive method with no base case
    public static void recursiveMethod()
    {
        recursiveMethod();// the method keeps calling itself
    }
}
```

```
Console ×
<terminated> StackOverflowErrorExample [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-20
StackOverflow errror caught:null
```

<> Invalid age exception

```java
package com.evergent.corejava.exceptionhandling;
class InvalidAgeException extends Exception
{
    public InvalidAgeException(String message)
    {
        super(message);
    }
}
public class UserDefinedExceptionDemo10 {
    public static void checkAge(int age) throws InvalidAgeException
    {
        if(age<18)
        {
            throw new InvalidAgeException("Age must be 18 or older");
        }
        else
        {
            System.out.println("Access granted- you are older enough");
        }
    }
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        try
        {
            checkAge(16);
        }
        catch(InvalidAgeException e)
        {
            System.out.println("Caught the exception:"+e.getMessage());
            System.out.println(e);
        }
        System.out.println("Program  continues after handling the exception");
    }
}
```

Console

```
<terminated> UserDefinedExceptionDemo10 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023
Caught the exception:Age must be 18 or older
com.evergent.corejava.exceptionhandling.InvalidAgeException: Age must be 18 or older
Program  continues after handling the exception
```

<>Insufficient fund exception

```java
package com.evergent.corejava.exceptionhandling;
class InsufficientFundsExceptions extends Exception
{
    public InsufficientFundsExceptions(String message)
    {
        super(message);
    }
}
public class UserDefinedExceptionsFunds11 {
    public static void withdraw(double amount) throws InsufficientFundsExceptions
    {
        double balance= 500.00;
        if(amount> balance)
        {
            throw new InsufficientFundsExceptions("Insufficient funds for withdraw");
        }
        else
        {
            System.out.println("Withdrawl successful");
        }
    }
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        try
        {
            withdraw(600.00);
        }
        catch(InsufficientFundsExceptions e)
        {
            System.out.println("caught InsufficientFundsExceptions:"+e.getMessage());
            System.out.println(e);
        }
        System.out.println("Program continues after handling the exception");
    }
}
```

Console

```
<terminated> UserDefinedExceptionsFunds11 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023
caught InsufficientFundsExceptions:Insufficient funds for withdraw
com.evergent.corejava.exceptionhandling.InsufficientFundsExceptions: Insufficient funds for withdraw
Program continues after handling the exception
```

<> Invalid score exception

```java
package com.evergent.corejava.exceptionhandling;
class InvalidScoreException extends RuntimeException
{
    public InvalidScoreException(String message)
    {
        super(message);
    }
}
public class UserDefinedUncheckedExceptionsDemo12 {
    public static void validateScore(int score)
    {
        if(score<0 || score>100)
        {
            throw new InvalidScoreException("Score must be between 0 and 100");
        }
        else
        {
            System.out.println("Score is valid");
        }
    }
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        try
        {
            validateScore(110);
        }
        catch(InvalidScoreException e)
        {
            System.out.println("caught the exception:"+e.getMessage());
            System.out.println(e);
        }
        System.out.println("Program flow continues");
    }
}
```

Console

<terminated> UserDefinedUncheckedExceptionsDemo12 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-

```
caught the exception:Score must be between 0 and 100
com.evergent.corejava.exceptionhandling.InvalidScoreException: Score must be between 0 and 100
Program flow continues
```

Day11:(21-08-24)
JAVA BEANS:

1. Java bean is a mechanism.
2. Java bean is lightweight.
3. All attributes are private.
4. Get/set methods are public.
5. Implements java. Io. Serialazable interface.
6. We can achieve tightly encapsulation through java beans.

<> Employee

```java
package com.evergent.corejava.javabeans;

import java.io.NotSerializableException;
public class Employee implements Serializable {
    private int eno;
    private String ename;
    private double sal;
    public void setEno(int eno) {
        this.eno=eno;
    }
    public void setEname(String ename) {
        this.ename=ename;
    }
    public void setSal(double sal) {
        this.sal=sal;
    }
    public int getEno() {
        return eno;
    }
    public String getEname() {
        return ename;
    }
    public double getSal() {
        return sal;
    }
}
```

<> initializing and retriving using setter and getter respectively

```java
package com.evergent.corejava.javabeans;

//initializing and retriving using setter and getter respectively
public class EmployeeImpl {
    public static void main(String[] args) {
        Employee emp=new Employee();
        emp.setEno(100);
        emp.setEname("Dhruti");
        emp.setSal(500.00);
        System.out.println("Employee no "+emp.getEno());
        System.out.println("Employee name "+emp.getEname());
        System.out.println("Employee sal "+emp.getSal());
    }
}
```

Console ×

<terminated> EmployeeImpl [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023-03\plugins\org.eclip

```
Employee no 100
Employee name Dhruti
Employee sal 500.0
```

<> Product

```java
package com.evergent.corejava.javabeans;

import java.io.NotSerializableException;
public class Product implements Serializable {
    private int pno;
    private String pname;
    private double price;
    public Product(int pno, String pname, double price) {
        this.pno=pno;

        this.pname=pname;

        this.price=price;
    }
    public int getPno() {
        return pno;
    }
    public String getPname() {
        return pname;
    }
    public double getPrice() {
        return price;
    }
}
```

<> initializing using constructor and retrving with getter

```java
package com.evergent.corejava.javabeans;

//initializing using constructor and retrving with getter
public class ProductImpl {
    public static void main(String[] args) {
        Product pd = new Product(100,"Dhruti",500.10);
        System.out.println("Product no "+pd.getPno());
        System.out.println("Product name "+pd.getPname());
        System.out.println("price "+pd.getPrice());

    }
}
```

Console

<terminated> ProductImpl (3) [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023-03\plug

```
Product no 100
Product name Dhruti
price 500.1
```

<> Student

```java
package com.evergent.corejava.javabeans;

import java.io.NotSerializableException;
public class Student implements Serializable {
    private int sno;
    private String sname;
    private double fee;
    public void setEno(int sno) {
        this.sno=sno;
    }
    public void setEname(String sname) {
        this.sname=sname;
    }
    public void setSal(double fee) {
        this.fee=fee;
    }
    public String toString() {
        return "Student no:"+sno+"/n Student name:"+sname+"/n student fee:"+fee;
    }
}
```

```java
package com.evergent.corejava.javabeans;

public class StudentImpl {
    public static void main(String[] args) {
        Student st=new Student();
        st.setEno(100);
        st.setEname("Dhruti");
        st.setSal(500.00);
        System.out.println(st);
    }
}
```

Console ×

<terminated> StudentImpl [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2(

```
Student no:100/n Student name:Dhruti/n student fee:500.0
```

Day12:(22-08-24)
COLLECTION FRAMEWORK:

<> ArrayList allows duplicate elements

```java
package com.evergent.corejava.collections;
import java.util.ArrayList;
// ArrayList allows duplicate elements
public class CF1_ArrayList {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        ArrayList myList= new ArrayList();
        myList.add(100);
        myList.add("hello");
        myList.add(21.1);
        myList.add(100);
        myList.add("Welcome");
        System.out.println(myList);
    }
}
```

```
Console ×
<terminated> CF1_ArrayList [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-20
[100, hello, 21.1, 100, Welcome]
```

<> HashSet does not allow duplicate elements

```java
package com.evergent.corejava.collections;
import java.util.HashSet;
public class CF2_HashSet {
//HashSet does not allow duplicate elements
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        HashSet mySet= new HashSet();
        mySet.add(100);
        mySet.add("100");
        mySet.add(100);
        mySet.add(21.3);
        mySet.add("hello");
        mySet.add("welcome");
        System.out.println(mySet);
    }
}
```

```
Console ×
<terminated> CF2_HashSet [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2
[100, 100, 21.3, hello, welcome]
```

<> Arraylist3

```java
package com.evergent.corejava.collections;
import java.util.ArrayList;
// ArrayList allows duplicate elements
public class CF3_ArrayList {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        ArrayList myList= new ArrayList();
        myList.add(100);
        myList.add("hello");
        myList.add(21.1);
        myList.add(100);
        myList.add("Welcome");
        System.out.println(myList);
        Iterator i=myList.iterator();
        while(i.hasNext())
        {
            System.out.println(i.next());
        }
    }
}
```

Console ×

<terminated> CF3_ArrayList [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-

```
[100, hello, 21.1, 100, Welcome]
100
hello
21.1
100
Welcome
```

<> Hashset 4

```java
package com.evergent.corejava.collections;
import java.util.HashSet;
public class CF4_HashSet {
//HashSet does not allow duplicate elements
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        HashSet mySet= new HashSet();
        mySet.add(100);
        mySet.add("100");
        mySet.add(100);
        mySet.add(21.3);
        mySet.add("hello");
        mySet.add("welcome");
        System.out.println(mySet);
        Iterator i=mySet.iterator();
        while(i.hasNext())
        {
            System.out.println(i.next());
        }
    }
}
```

Console ×

<terminated> CF4_HashSet [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-

```
[100, 100, 21.3, hello, welcome]
100
100
21.3
hello
welcome
```

<> TreeSet

```java
package com.evergent.corejava.collections;
import java.util.TreeSet;
public class CF5_TreeSet {

	public static void main(String[] args) {
		// TODO Auto-generated method stub
		TreeSet mySet= new TreeSet();
		mySet.add(100);
		mySet.add(40);
		mySet.add(60);
		mySet.add(21);
		//mySet.add("hello");
		//mySet.add("welcome");
		System.out.println(mySet);
		Iterator i=mySet.iterator();
		while(i.hasNext())
		{
			System.out.println(i.next());
		}
	}
}
```

```
Console ×
<terminated> CF5_TreeSet [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse
[21, 40, 60, 100]
21
40
60
100
```

<> ListIterator

```java
package com.evergent.corejava.collections;
import java.awt.List;
public class CF6_ListIterator {
	public static void main(String[] args) {
		// TODO Auto-generated method stub
		ArrayList list= new ArrayList();
		list.add("Dhruti");
		list.add("Vinay");
		list.add("Roshu");
		list.add("Hari");
		ListIterator li=list.listIterator();
		li.add("Welcome");
		while(li.hasNext())
		{
			String s= (String) li.next();
			System.out.println(s);
			if(s.equals("Roshu"))
			    li.remove();
			}
		while(li.hasPrevious())
			System.out.println(li.previous());
	}
}
```

```
Console ×
<terminated> CF6_ListIterator [Java Application] C:\Users\dhruti.guthikonda\Desk
Dhruti
Vinay
Roshu
Hari
Hari
Vinay
Dhruti
Welcome
```

<> Vector

```java
package com.evergent.corejava.collections;
import java.util.Vector;
public class CF7_Vector {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Vector v= new Vector();
        v.add("Hello");
        v.add(100);
        v.add(45.5);
        System.out.println(v);
        Enumeration e=v.elements();
        while(e.hasMoreElements())
        {
            System.out.println(e.nextElement());
        }

    }

}
```

Console ×

<terminated> CF7_Vector [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclip

```
[Hello, 100, 45.5]
Hello
100
45.5
```

<> LinkedListMethods

```java
package com.evergent.corejava.collections;
import java.util.LinkedList;
public class CF8_LinkedListMethods {
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        //create a LinkedList of Strings
        LinkedList fruits=new LinkedList();
        //add elements to linkedlist
        fruits.add("apple");
        fruits.add("Banana");
        fruits.add("Cherry");
        fruits.add("Kiwi");
        //displaying the linkedlist
        System.out.println("Initial list:"+fruits);
        //Add an element at te begining
        fruits.addLast("Mango");
        System.out.println("after adding element:"+fruits);
        fruits.addLast("Orange");
        System.out.println("after adding last element:"+fruits);
        //access elements by index
        System.out.println("Element at index2:"+fruits.get(2));
        //Remove the first and last elements
        fruits.removeFirst();
        System.out.println("after removing first:"+fruits);
        fruits.removeLast();
        System.out.println("after removing last"+fruits);
        //check if the linkedlist contains a specific element
        System.out.println("Contains Banana?"+fruits.contains("Banana"));
        //Remove an element by value
        fruits.remove("Banana");
        System.out.println("After deletion:"+fruits);
        //get size of the ll
        System.out.println("size of ll:"+fruits.size());
        //clear the ll
        fruits.clear();
        System.out.println(fruits);
    }
}
```

```
Console ×
<terminated> CF8_LinkedListMethods [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023-03\p
Initial list:[apple, Banana, Cherry, Kiwi]
after adding element:[apple, Banana, Cherry, Kiwi, Mango]
after adding last element:[apple, Banana, Cherry, Kiwi, Mango, Orange]
Element at index2:Cherry
after removing first:[Banana, Cherry, Kiwi, Mango, Orange]
after removing last[Banana, Cherry, Kiwi, Mango]
Contains Banana?true
After deletion:[Cherry, Kiwi, Mango]
size of ll:3
[]
```

<> CF9

```java
package com.evergent.corejava.collections;
import java.util.ArrayList;
class Book5
{
    String name;
    public Book5(String name) {
        this.name=name;
    }
    public String toString()
    {
        return name;
    }
}
public class CF9_ArrayList_BookObject {
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Book5 b1=new Book5("Core Java");
        Book5 b2=new Book5("Let us c");
        Book5 b3=new Book5("Java index Book");
        Book5 b4=new Book5("Java interview book");
        ArrayList mylist=new ArrayList();
        mylist.add(b1);
        mylist.add(b2);
        mylist.add(b3);
        mylist.add(b4);
        System.out.println(mylist);
    }
}
```

```
Console ×
<terminated> CF9_ArrayList_BookObject [Java Application] C:\Users\dhruti.guthikonda\Desktop
[Core Java, Let us c, Java index Book, Java interview book]
```

<> ArrayList_Generics

```java
package com.evergent.corejava.collections;
import java.util.ArrayList;
public class CF10_ArrayList_Generics {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        ArrayList<Integer> mylist=new ArrayList<Integer>();
        mylist.add(100);
        mylist.add(90);
        mylist.add(45);
        mylist.add(100);
        System.out.println(mylist);
        Iterator i=mylist.iterator();
        while(i.hasNext())
        {
            System.out.println(i.next());
        }
    }
}
```

```
Console ×
<terminated> CF10_ArrayList_Generics [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-202
[100, 90, 45, 100]
100
90
45
100
```

<> HashSet_Generics

```java
package com.evergent.corejava.collections;
import java.util.*;
public class CF11_HashSet_Generics {


    public static void main(String[] args) {
        // Create a HashSet of Strings
        HashSet<String> myset = new HashSet<>();
        myset.add("Dhruti");
        myset.add("Vinay");
        myset.add("Roshu");
        myset.add("hari");

        // Print the HashSet
        System.out.println(myset);

        // Create an Iterator for the HashSet
        Iterator<String> i = myset.iterator();

        // Iterate through the HashSet and print each element
        while (i.hasNext()) {
            System.out.println(i.next());
        }
    }
}
```

```
Console ×
<terminated> CF11_HashSet_Generics [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\
[hari, Dhruti, Roshu, Vinay]
hari
Dhruti
Roshu
Vinay
```

Day13:(23-08-24)
WRAPPER CLASS:

<> Wrapper class demo1

```java
package com.evergent.corejava.wrapperclasses;

public class WrapperClassesDemo1 {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        //Auto Boxing
        int a=10;
        Integer i1=new Integer(a);// storing int value in object
        System.out.println(i1);
        // Unboxing
        int a1=i1.intValue();
        System.out.println(a1);

    }

}
```

Console ×

<terminated> WrapperClassesDemo1 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03

```
10
10
```

<> Wrapper class Demo2

```java
package com.evergent.corejava.wrapperclasses;

public class WrapperClassesDemo2 {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        int i1=100;
        Integer t1= new Integer(i1);
        int i2=t1.intValue();
        double d1=999.76;
        Double t2= new Double(d1);
        double d2= t2.doubleValue();
        byte b1=10;
        Byte t3= new Byte(b1);
        byte b2=t3.byteValue();
        //integer values
        System.out.println("int value is:"+i1);
        System.out.println("int object value is:"+t1);
        System.out.println("After converting int obj value to primitive is:"+i2);
        //Double value
        System.out.println("double value is:"+d1);
        System.out.println("double object value is:"+t2);
        System.out.println("After converting double obj value to primitive is:"+d2);
        //Bye value
        System.out.println("byte value is:"+b1);
        System.out.println("byte object value is:"+t3);
        System.out.println("After converting byte obj value to primitive is:"+b2);
    }
}
```

Console ×

<terminated> WrapperClassesDemo2 [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclip

```
int value is:100
int object value is:100
After converting int obj value to primitive is:100
double value is:999.76
double object value is:999.76
After converting double obj value to primitive is:999.76
byte value is:10
byte object value is:10
After converting byte obj value to primitive is:10
```

<> Autoboxing_Unboxing

```java
package com.evergent.corejava.wrapperclasses;

public class Autoboxing_Unboxing3 {

	public static void main(String[] args) {
		// TODO Auto-generated method stub
		int a=10;
		Integer i= new Integer(a);
		//unboxing the object
		int i1=i.intValue();
		System.out.println("value of i:"+i);
		System.out.println("value of i1:"+i1);
		//autoboxing of character
		char ch='a';
		Character ch1=new Character(ch);
		// or Character ch1=ch;
		//autounboxing
		char ch2=ch1.charValue();
		System.out.println("value of ch1:"+ch1);
		System.out.println("value of ch2:"+ch2);

	}

}
```

```
Console ×
<terminated> Autoboxing_Unboxing3 [Java Application] C:\Users\dhruti.guthikonda\D
value of i:10
value of i1:10
value of ch1:a
value of ch2:a
```

<> WrapperClassesDemo4_ArrayList

```java
public class WrapperClassesDemo4_ArrayList {

	public static void main(String[] args) {
		// TODO Auto-generated method stub
		//JDK 1.44 -boxing and unboxing done by developers
		int a=100;
		Integer i1=new Integer(a);
		ArrayList mylist=new ArrayList();
		mylist.add(i1);
		System.out.println(mylist);
		Integer i2=new Integer(200);
		mylist.add(i2);
		System.out.println(mylist);
		mylist.add(new Integer(45));
		System.out.println(mylist);
		System.out.println(mylist.get(1));// gives the value from the index provided
		//JDK 1.5 - compiler/jvm automatically does boxing and unboxing.
		ArrayList mylist1= new ArrayList();
		mylist1.add(100);// boxing
		System.out.println(mylist.get(0));//unboxing

	}

}
```

```
Console ×
<terminated> WrapperClassesDemo4_ArrayList [Java Application] C:\Users\dhruti.guthikonda\Desktop\eclipse-2023-03\eclipse-2023-03\pl
[100]
[100, 200]
[100, 200, 45]
200
100
```