Day 7

Constructor Overloading

Constructor can be repeated more times by arguments should be added in duplicate.

Eg:
```
student(){
    this.rno = 13;
    this.name = "Priya";
}
student(int roll, String name){
    this.rno = roll;
    this.name = name;
}
```
→ We can change the value assign some other value.

Memory allocation of 'new' Keyword:

Eg.
```
Student one = new Student();
Student two = one;
one.name = "Hi all!";
System.out.print(two.name);
```
If we change one value then it will give changes in two also

Wrapper Class

Convert primitive datatypes into object.
Collections only allowed obj data

Primitive Datatypes Can be written as Wrapper Class

Eg:

Primitive Datatypes	Wrapper Class
int	Integer
float	Float
boolean	Boolean
Char	Character
~~B~~ long	Long
short	Short
double	Double
byte	Byte

Eg:

```
public class Practice{
    public static void main
                        (String[] args){
        int i = 10;
        Integer data = new Integer(i);
        int j;
        j = data.intValue();
        System.out.println(j);
    }
}
```

primitive data to obj data

Autoboxing

Unboxing

O/p:

10

Eg:

```
public class WrapperEg{
    public static void main
                        (String[] args){
        int a=10, b=20;
        Integer num=45;
        swap(a,b);
        System.out.println(a+" "+b);
    }
    static void swap(Integer a,
                     Integer b){
        int temp =a;
        a=b;
        b=temp;
    }
}
```

Swap cannot happen

O/p: 10
20

Final Keyword

→ Variable

→ Method

→ Class

Final Variable: Add final keyword in front of variable

Eg: final float pi = 3.14f;   Constant value, If we change get compilation error

Blank final variable:

→ Variable does not initialize over time called blank final variable.

Eg:

```
class Demo{
    final float pi;
    Demo(float p){
        pi=p;
    }
    //code
}
```

Must be initialized in Constructor otherwise get compilation error

Final Method:

Cannot override a final method

Syn final void display(){
}

Eg:

```
class SeniorProgrammer{
private int salary;
int experience;
final void calcSalary(){
salary = 5000+ (2000* experience);
}
class Programmer extends SeniorProgrammer{
void calcSalary(){
//error
}
}
Programmer p1 = new Programmer();
p1.calcSalary();
```

Cannot override

Can call obj method but cannot override

Final Class:

Cannot extended

```
final class Confidential{
  //code
}
Class A extends Confidential{
  :
}
```

Cannot Happen

Compilation Error

NOTE:

- Final class cannot be extended
- Final Method cannot be override

- Cannot change value of final variable
- Cannot declare a constructor as final

Vowel Count Program

```
public class VowelCount {
   private String str;
   public VowelCount(String str){
       this.str = str;
   }
   public int countvowels(){
       int count = 0;
       for(char c: str.toCharArray()){
          if(c=='a'||c=='e' || c=='i'||c=='o'|| c=='u'){
              Count++;
          }
       }
       return count;
   }
   public static void main(String[] args){
       VowelCount counter = new VowelCounter("Priya");
       int vowel = counter.countvowels();
       System.out.println("Number of Vowels:" + vowel);
   }
}
```

O/p:

```
Number of Vowels: 2
```

In this, we count vowel char & represent it as a int value.