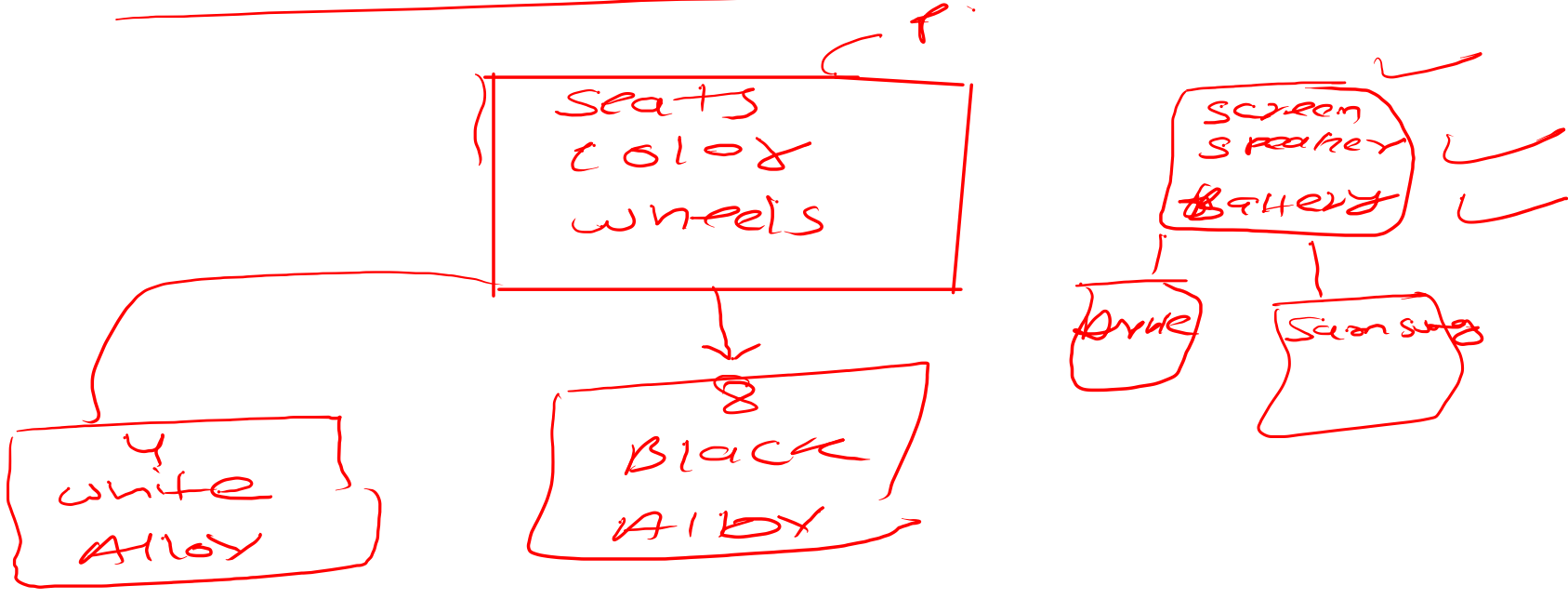
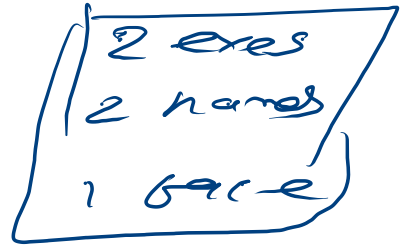
Applications

(1) Class & object



Class → A Class is a template.

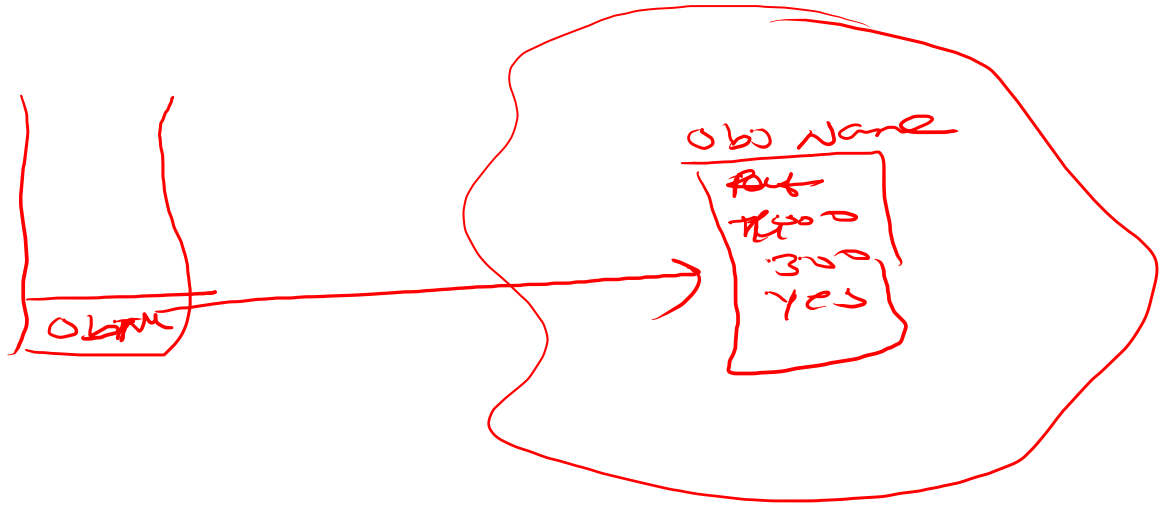
object → A object is a Instance of Class.



Obj
2 eyes
2 hand
1 face

```
Class Emp {
    String name;
    int age;
    int Empid;
    String address
    int phone;
}
```

Register objName = new Register()



Constructor

+ methods & functions

```
void greeting() {

}
```

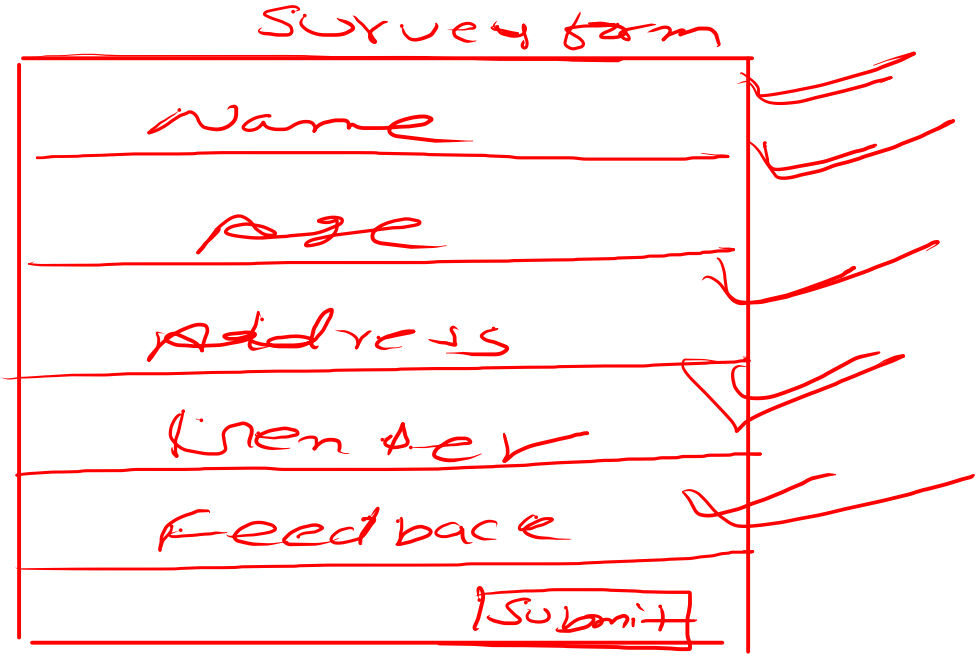
A) greeting ;

B) greeting(); ✓

1. Constructor are special type of methods.
2. Constructor don't have a return type.
3. Constructor Name Same as Class Name.
4. Constructor are automatically Invoked.

1. Default constructor

2. Parameterized constructor



```java
import java.io.*;
import java.util.*;

public class Solution {

    static class RaftaarMusic{
        String name = "Raftaar";
        int post = 340;
        int totalReels = 400;
        String bluetick = "Yes";
        int followers = 7;
        int following = 200;
        String category = "Rapper";
        String Gender = "M";
    }

    static class RachelGupta{
       String name = "Rachel Gupta";
        int post = 550;
        int totalReels = 350;
        String bluetick = "Yes";
        int followers = 15;
        int following = 10;
        String category = "Artist";
        String Gender = "M";

    }
    static class NikhilChinapa{
        String name = "Nikhil Chinapa";
        int post = 500;
        int totalReels = 220;
        String bluetick = "Yes";
        int followers = 5;
        int following = 210;
        String category = "VJ";
        String Gender = "M";

    }
```

```
    static class NikhilChinapa{
        String name = "Nikhil Chinapa";
        int post = 500;
        int totalReels = 220;
        String bluetick = "Yes";
        int followers = 5;
        int following = 210;
        String category = "VJ";
        String Gender = "M";

    }
    static class Baseer_bob{
        String name = "Baseer";
        int post = 500;
        int totalReels = 120;
        String bluetick = "No";
        int followers = 2;
        int following = 230;
        String category = "VJ";
        String Gender = "M";
    }
```

```
public static void main(String[] args) {
    /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution.
    RaftaarMusic raaftar = new RaftaarMusic();
    RachelGupta rachel = new RachelGupta();
    NikhilChinapa nikhil = new NikhilChinapa();
    Baseer_bob baseer = new Baseer_bob();

    System.out.println(raaftar.name);
    System.out.println(raaftar.post);
    System.out.println(raaftar.totalReels);
    System.out.println(raaftar.bluetick);
    System.out.println(raaftar.followers);
    System.out.println(raaftar.following);
    System.out.println(raaftar.category);
    System.out.println(raaftar.Gender);

    System.out.println(rachel.name);
    System.out.println(rachel.post);
    System.out.println(rachel.totalReels);
    System.out.println(rachel.bluetick);
    System.out.println(rachel.followers);
    System.out.println(rachel.following);
    System.out.println(rachel.category);
    System.out.println(rachel.Gender);

    System.out.println(nikhil.name);
    System.out.println(nikhil.post);
    System.out.println(nikhil.totalReels);
    System.out.println(nikhil.bluetick);
    System.out.println(nikhil.followers);
    System.out.println(nikhil.following);
    System.out.println(nikhil.category);
    System.out.println(nikhil.Gender);
```

```
    System.out.println(baseer.name);
    System.out.println(baseer.post);
    System.out.println(baseer.totalReels);
    System.out.println(baseer.bluetick);
    System.out.println(baseer.followers);
    System.out.println(baseer.following);
    System.out.println(baseer.category);
    System.out.println(baseer.Gender);
```

```
public static int sum(int a, int b){
    return a + b;
}
}
public sum() {
    int a = 5;
    int b = 10;
    sum(a, b);
}
```