**Abstract Class**

**1. Create an abstract class with abstract and non-abstract methods.**

abstract class Bike{

abstract void run();

}

class Honda4 extends Bike{

void run(){System.out.println("running safely");}

public static void main(String args[]){

Bike obj = new Honda4();

obj.run();

}

}

abstract class AbstractDemo {

private int i = 0;

public void display() {

System.out.print("Welcome to Tutorials Point");

}

}

public class InheritedClassDemo extends AbstractDemo {

public static void main(String args[]) {

AbstractDemo demo = new InheritedClassDemo();

demo.display();

}

}

**2. Create a sub class for an abstract class. Create an object in the child class for the abstract class and access the non-abstract methods**

abstract class Language {

// method of abstract class

public void display() {

System.out.println("This is Java Programming");

}

}

class Main extends Language {

public static void main(String[] args) {

Main obj = new Main();

obj.display();

}

}

abstract class Animal {

abstract void makeSound();

public void eat() {

System.out.println("I can eat.");

}

}

class Dog extends Animal {

public void makeSound() {

System.out.println("Bark bark");

}

}

class Main {

public static void main(String[] args) {

Dog d1 = new Dog();

d1.makeSound();

d1.eat();

}

}

**3. Create an instance for the child class in child class and call abstract methods**

abstract class Animal {

abstract static Animal fromInput(String input); // <- error

static List<Animal> makeFive() {

List<Animal> animals = new ArrayList<Animal>();

animals.add(Animal.fromInput("Input 1"));

animals.add(Animal.fromInput("Input 2"));

animals.add(Animal.fromInput("Input 3"));

animals.add(Animal.fromInput("Input 4"));

animals.add(Animal.fromInput("Input 5"));

return animals;

}

}

class Dog extends Animal {

@Override static Dog fromInput(String input) {

Dog dog = new Dog();

return dog;

}

}

class Cat extends Animal {

@Override static Cat fromInput(String input) {

Cat cat = new Cat();

return cat;

}

}

**4. Create an instance for the child class in child class and call non-abstract methods**

public abstract class Test {

void t1()

{

System.out.println("super");

}

}

public class concret extends Test{

void t1()

{

System.out.println("child");

}

void t2()

{

System.out.println("child2");

}

}

public class run {

public static void main(String[] args) {

Test t=new concret();

t.t1();

}

}