Lesson 14
Inheritance(Vorislik)

Inheritance(Vorislik)

Vorislik(inheritence) – bu bir klass ikkinchi klassning maydon va metodlarini ishlata olish qobilyati hisoblanadi.Ya`ni ikki sinf bo`lib:

Ota(super)klass-xususiyatlari meros qilinib olinadigan sinf.

Bola(sub) klass-boshqa sinfdan meros oladigan sinf.

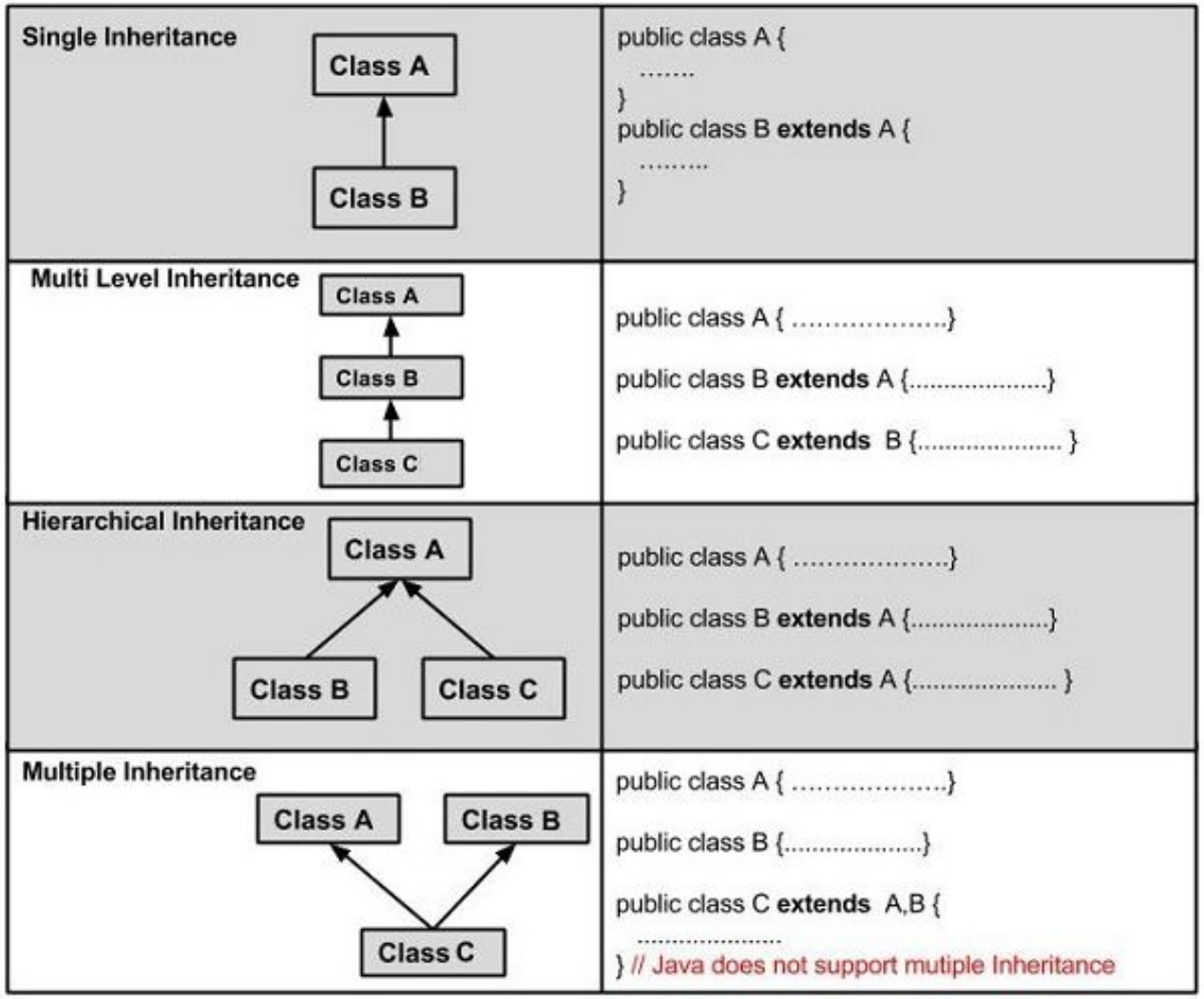
Extends(kengaytirish)-bu sinf xususiyatlarini meros qilib olish uchun ishlatiladigan kalit so'z.

Inheritance(Vorislik)


```
public class Super {
        //maydonlar
    //metodlar
}

public class Sub extends Super {
        //maydonlar
    //metodlar
}
```

Inheritance(Vorislik)



Inheritance(Vorislik)


```
class Vehicle {
  protected String brand = "Ford";        // Vehicle attribute
  public void honk() {                    // Vehicle method
    System.out.println("Tuut, tuut!");
  }
}

class Car extends Vehicle {
  private String modelName = "Mustang";    // Car attribute
  public static void main(String[] args) {
    // Create a myCar object
    Car myCar = new Car();
    // Call the honk() method (from the Vehicle class) on the myCar object
    myCar.honk();

    // Display the value of the brand attribute (from the Vehicle class) and the value of the modelName from the Car class
    System.out.println(myCar.brand + " " + myCar.modelName);
  }
}
```

Inheritance(Vorislik)


```java
package com.company;

class Bicycle {
    // the Bicycle class has two fields
    public int gear;
    public int speed;

    // the Bicycle class has one constructor
    public Bicycle(int gear, int speed) {
        this.gear = gear;
        this.speed = speed;
    }

    // the Bicycle class has three methods
    public void applyBrake(int decrement) {
        speed -= decrement;
    }

    public void speedUp(int increment) {
        speed += increment;
    }

    // toString() method to print info of Bicycle
    public String toString() {
        return ("No of gears are " + gear
                + "\n"
                + "speed of bicycle is " + speed);
    }
}
```

```java
// derived class
class MountainBike extends Bicycle {

    // the MountainBike subclass adds one more field
    public int seatHeight;

    // the MountainBike subclass has one constructor
    public MountainBike(int gear, int speed,
                        int startHeight) {
        // invoking base-class(Bicycle) constructor
        super(gear, speed);
        seatHeight = startHeight;
    }

    // the MountainBike subclass adds one more method
    public void setHeight(int newValue) {
        seatHeight = newValue;
    }

    // overriding toString() method
    // of Bicycle to print more info
    @Override
    public String toString() {
        return (super.toString() +
                "\nseat height is " + seatHeight);
    }

}
```

```java
// driver class
class Test {

    public static void main(String args[]) {


        MountainBike mb = new
MountainBike(3, 100, 25);
        System.out.println(mb.toString());


    }
}
```

Inheritance(Vorislik)

Super – bu kalit so`z orqali bola klass ota klassning maydon va metodlarini o`zining ichida ishlatish huquqiga ega bo`la oladi

Super() – vorislikda bu orqali ota klass konstruktori ishga tushiriladi.

```
class Super {
    int a;
    int b;

    public Super(int a, int b) {
        this.a = a;
        this.b = b;
    }
    public void show() {
        System.out.println("Hello world");
    }
}
```

```
class Sub extends Super {

    public Sub(int a, int b) {
        super(a, b);
    }

    public void showAB(){
        super.show();
        System.out.println(super.a+" "+super.b);
    }
}
```

Inheritance(Vorislik) – Override

Override – bu ota klassning metodlarini qayta yozish hisoblanadi.

Override qilingan metodlarning murojat turlarinini bolasida o`zgartirish huquqiga ega bo`lishimiz mumkin.

```
class Super {
    protected void show(){
        System.out.println("Super");
    }
}

class Sub extends Super {

    @Override
    public void show() {
        super.show();
    }
}
```

Inheritance(Vorislik) – Override

Java da barcha klasslar Object klasidan voris olingan hisoblanadi. Bu klassning bir necha maydonlari mavjut:

```
public int hashCode() {
    return super.hashCode();
}

public boolean equals(Object obj) {
    return super.equals(obj);
}

\protected Object clone() throws CloneNotSupportedException {
    return super.clone();
}

public String toString() {
    return super.toString();
}

protected void finalize() throws Throwable {
    super.finalize();
}
```

Inheritance(Vorislik) – Override


```
class Parent {

    private int age;

    public int getAge() {
        return age;
    }

    public void setAge(int age) {
        this.age = age;
    }

}
```

```
class A extends Parent {

}

class B extends Parent {

}

class C extends Parent {

}
```

```
class Test {
    public static void main(String[] args) {
        Scanner scanner=new Scanner(System.in);
        Parent  p[]=new  Parent[3];
        p[0]=new A();
        p[1]=new B();
        p[2]=new C();
        for (int i = 0; i < p.length; i++) {
            p[i].setAge(scanner.nextInt());
        }
        for (int i = 0; i < p.length; i++) {
            System.out.println(p[i].getAge());
        }
    }
}
```

The end