

1 الوراثة

في Java، من الممكن أن ترث Attributes و methods من كلاس إلى كلاس آخر. مفهوم Inheritance يندرج تحته ثلاث مفاهيم أساسية:

- الكلاس الفرعي subclass (الطفل) - الكلاس الذي يرث من كلاس آخر
 - الكلاس الأساسي superclass (الأصل) - الكلاس الموروث منه
 - لنرث صفات كلاس نستخدم كلمة extends
- في المثال أدناه، ترث كلاس Car (الكلاس الفرعي) attributes و methods من فئة Vehicle (الكلاس الأساسي):

```
class Vehicle{
    void moving() {
        System.out.println("The Vehicle is moving");
    }
}

class Car extends Vehicle{
}
```

كما في المثال الأعلى يمكننا استخدام جميع methods الخاصة بكلاس Vehicle عند انشاء object من نوع Car:

```
public class Main {
    public static void main(String[] args) {
        Car car1 = new Car();
        car1.moving();
    }
}
```

المخرجات:

```
The Vehicle is moving
```

رغم اننا لم نعرف اي method باسم moving في الكلاس الخاص ب Car لكننا استطعنا استخدام دالة moving والسبب لأنها موروثة من كلاس Vehicle ونستطيع استخدامها.

1.2.1 استخدام super

عند وجود Constructor في الكلاس الأساسي وأردنا وراثة هذا الكلاس فإننا سنحتاج الى استخدام كلمة super في Construction الخاص بالكلاس الفرعي كما في المثال التالي:

```
class Vehicle{
    String speed;
    int yearOfManufacture;

    public Vehicle(String speed, int yearOfManufacture) {
        this.speed = speed;
        this.yearOfManufacture = yearOfManufacture;
    }

    void moving() {
        System.out.println("The Vehicle is moving");
    }
}
```

```
    }  
}  
  
class Car extends Vehicle{  
    String company;  
    public Car(String speed, int yearOfManufacture, String company)  
    {  
        super(speed, yearOfManufacture);  
        this.company = company;  
    }  
}
```