

Covariant return types

Method overriding is a mechanism for providing new behaviour for the superclass method in a subclass method.

The subclass method can return the same type as the superclass method or a subtype of this return type. This feature is known as the **covariant return type**.

Covariant return type allows you to narrow (but not widen) the return type of the overridden method, that is to make it more specific.

```
class SuperType { }

class SubType extends SuperType { }

class A {

    public SuperType getType() {
        return new SuperType();
    }
}

class B extends A {

    @Override
    public SubType getType() {
        return new SubType();
    }
}
```

The next example, though, doesn't compile:

```
class A {

    public SubType getType() {
        return new SubType();
    }
}

class B extends A {

    @Override
    public SuperType getType() {
        return new SuperType();
    }
}
```

Summary

Remember, when overriding the return type of a subclass method can be the same type or a subclass of the return type of a superclass method. Covariance is always changing down the hierarchy.

It is also important to note that the covariant return types are possible only for non-primitive return types.