Explicando o Override

Estou com algumas dúvidas quanto ao @override, li em algum lugar e me lembro vagamente sobre a questão da reescrita, mas o que é isso? E para que serve? Como se aplica em um código **JAVA**. Existe em alguma outra linguagem?

Ele é uma forma de garantir que você está sobrescrevendo um método e não criando um novo.

Vamos supor que você criou uma classe com um método de imprimir:

public class SuperClasse {

public void imprime() {

System.out.println("imprime");

}

}

Daí você fez uma classe que estende essa classe, e você quer mudar o que será impresso quando o método for chamado:

public class MinhaClasse extends SuperClasse {

public void imprime() {

System.out.println("imprime diferente");

}

}

Você sobrescreveu corretamente o método imprime() e o código acima funcionará sem maiores problemas.

Entretanto um belo dia você resolveu mudar o nome do método na SuperClasse de imprime() para imprimir():

public class SuperClasse {

public void imprimir() {

System.out.println("imprime");

}

}

Se você não se lembrar da classe que estendeu a SuperClasse você terá um método imprime() nela que **não está sobrescrevendo o método na SuperClasse**. Você na verdade está chamando um método novo, que se chama imprime().

O que aconteceria se você tivesse usado o annotation @Override ?

Esse seria seu código a princípio para as duas classes:

public class SuperClasse {

public void imprime() {

System.out.println("imprime");

}

}

public class MinhaClasse extends SuperClasse {

@Override

public void imprime() {

System.out.println("imprime diferente");

}

}

Até o momento, nada de novo. Entretanto quando você alterar o seu método de imprime() para imprimir() você não mais conseguirá compilar seu código pois o @Override perceberá que você não está sobrescrevendo coisa nenhuma, pois não existe mais nenhum método imprime() na SuperClasse.

Você receberá o seguinte erro:

MinhaClasse must override or implement a supertype method

Em tradução (bem) livre:

MinhaClasse deve sobrescrever ou implementar um método da sua super classe

Toda linguagem orientada a objetos permite a sobrescrita de métodos da super classe pela classe filha. Entretanto cada linguagem de programação usa seus próprios meios para lidar com essa sobrescrita. O Java optou por usar o annotation @Override para os desenvolvedores que quiserem a segurança citada no decorrer da resposta, entretanto, nada obriga o uso desse annotation.

O C# também possui tal funcionalidade, entretanto ele não usa annotation, ele incorpora a palavra reservada override na declaração do método:

public override int Area()

{

return side \* side;

}

É impossível listar todas as linguagens de programação que fazem uso do override pois acredite se quiser, existem milhares (literalmente!) de linguagens por aí. Se quiser uma lista um pouco mais completa veja: [Wikipedia - Method overriding](http://en.wikipedia.org/wiki/Method_overriding)

<https://pt.stackoverflow.com/questions/22913/qual-a-finalidade-da-override/22922>