

PROGRAMAÇÃO EM JAVA

MÓDULO 4

A PALAVRA RESERVADA “SUPER”

Praticando 1

A palavra reservada “super” é utilizada para chamar o método da classe mãe dentro da classe filha. Imagine o cenário em que o Gerente receberia a mesma bonificação do funcionário, isto é 10% do salário somado ao valor de 300 reais. Utilizando a palavra reservada “super”, nosso método `BonificacaoAnual()` poderia ficar assim:

```
package Missao6;  
public class Gerente extends Funcionario  
{  
    .....  
    public double BonificacaoAnual()  
    {  
        double SalarioComBonificacao = super.BonificacaoAnual();  
        SalarioComBonificacao = SalarioComBonificacao + 300;  
        return SalarioComBonificacao;  
    }  
    ....  
}
```

- Note que, neste caso, o método **BonificacaoAnual()** da classe Gerente está chamando o método **BonificacaoAnual()** da classe Funcionário e armazenando o resultado na variável **SalarioComBonificacao**.
- Na linha seguinte, o valor da variável **SalarioComBonificacao** é acrescido de 300 reais e só depois este valor é devolvido à classe principal.

Como você pode notar, a utilização da palavra reservada “super” permite economizar linhas de código e retrabalho!

ATENÇÃO

Outro item importante, quando trabalhamos com o conceito de Override (Sobrescrita de métodos) é a anotação **@override**. Ela garante que você está sobrescrevendo um método e não criando um outro.

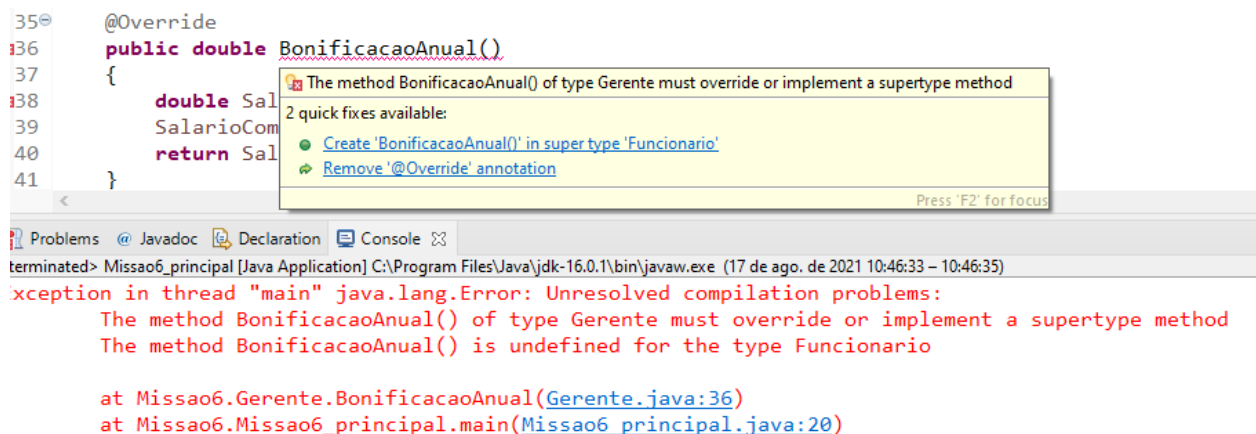
Praticando 2

Para exemplificar, vamos usar o mesmo método `BonificacaoAnual()` trabalhado acima. Deste modo, altere o código acima para que inclua a anotação `@Override`, da seguinte maneira:

```
package Missao6;
public class Gerente extends Funcionario
{
    ....
    @Override
    public double BonificacaoAnual()
    {
        double SalarioComBonificacao = super.BonificacaoAnual();
        SalarioComBonificacao = SalarioComBonificacao + 300;
        return SalarioComBonificacao;
    }
    ....
}
```

A introdução dessa anotação (anotation) garante que se o método `BonificacaoAnual()` da classe mãe tiver seu nome alterado (por qualquer motivo), o compilador não vai executar o programa e apresentará um erro.

Se você alterar o nome do método `BonificaAnual()` da classe `Funcionario` para `BonificaAnual1()` e tentar rodar o programa, o resultado será como mostrado na figura abaixo:



Erro de Override

O erro ocorreu dentro da classe `Gerente`, na linha 36 e ele relata que: “O método `BonificacaoAnual()` da classe `Gerente` deve sobrescrever ou implementar o método da superclasse. Mas como a superclasse (`Funcionario`) não possui mais um método com esse nome, o erro é apresentado.