

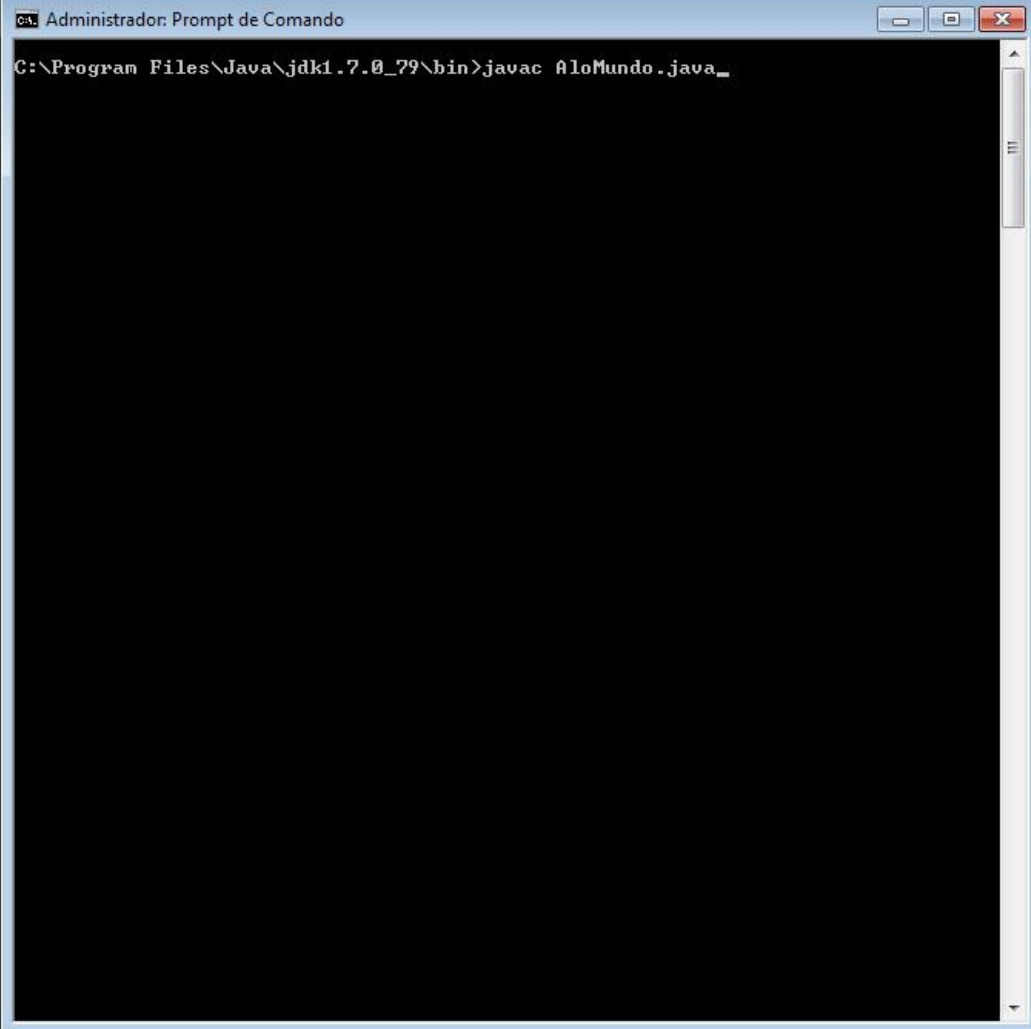
Como Identificar e Corrigir Erros na Compilação em Java

Administrador: Prompt de Comando

C:\Program Files\Java\jdk1.7.0_79\bin>_

Vamos ver, agora, um exemplo de compilação com erro:

Vale lembrar que é possível **PAUSAR** ou **RETROCEDER** qualquer explicação que achar conveniente.



```
Administrador: Prompt de Comando
C:\Program Files\Java\jdk1.7.0_79\bin>javac AloMundo.java_
```

```
Administrator: Prompt de Comando

C:\Program Files\Java\jdk1.7.0_79\bin>javac AloMundo.java
AloMundo.java:5: error: cannot find symbol
    System.out.println("Ola " + MSG);
                               ^
    symbol:   variable MSG
    location: class AloMundo
1 error

C:\Program Files\Java\jdk1.7.0_79\bin>
```

```

C:\Program Files\Java\jdk1.7.0_79\bin>javac AloMundo.java
AloMundo.java:5: error: cannot find symbol
    System.out.println("Ola " + MSG);
                               ^
symbol:   variable MSG
location: class AloMundo
1 error
C:\Program Files\Java\jdk1.7.0_79\bin>
    
```

Esse erro ocorreu por causa da alteração em apenas uma variável (ou um elemento) do código para a simulação de erros de compilação, e essa simples mudança gerou esse 1 error na variável MSG da classe AloMundo.

```

C:\Program Files\Java\jdk1.7.0_79\bin>javac AloMundo.java
AloMundo.java:5: error: cannot find symbol
    System.out.println("Ola " + MSG);
                               ^
    symbol:   variable MSG
    location: class AloMundo
1 error
C:\Program Files\Java\jdk1.7.0_79\bin>
    
```

Você pode estar se perguntando como uma simples variável ou um elemento qualquer no código pode causar um erro?

```

C:\Program Files\Java\jdk1.7.0_79\bin>javac AloMundo.java
AloMundo.java:5: error: cannot find symbol
    System.out.println("Ola " + MSG);
                        ^
symbol:   variable MSG
location: class AloMundo
1 error
C:\Program Files\Java\jdk1.7.0_79\bin>
    
```

Podemos garantir que apenas uma variável ou um elemento qualquer no código pode gerar erro na compilação.

Tudo depende da **quantidade de relacionamentos e envolvimento** no código onde o elemento ou a variável foram mencionados ou aplicados.

```

C:\Program Files\Java\jdk1.7.0_79\bin>javac AloMundo.java
AloMundo.java:5: error: cannot find symbol
    System.out.println("Ola " + MSG);
                        ^
symbol:   variable MSG
location: class AloMundo
1 error
C:\Program Files\Java\jdk1.7.0_79\bin>
    
```

Desse modo, poderemos ter erros simples de serem resolvidos e também erros muito complexos, que irão exigir horas de apuração para encontrar e resolver o problema.


```

C:\Program Files\Java\jdk1.7.0_79\bin>javac AloMundo.java
AloMundo.java:5: error: cannot find symbol
    System.out.println("Ola " + MSG);
                        ^
    symbol:   variable MSG
    location: class AloMundo
1 error
C:\Program Files\Java\jdk1.7.0_79\bin>
  
```

Como seu olhar para o código estará "viciado", é possível que você não visualize e não detecte o erro imediatamente ou após horas de conferência.

Nesse caso, você pode precisar do auxílio de outro profissional com a cabeça mais "fresca", mais neutro, e que não esteja envolvido com o código.

```

C:\Program Files\Java\jdk1.7.0_79\bin>javac AloMundo.java
AloMundo.java:5: error: cannot find symbol
    System.out.println("Ola " + MSG);
                        ^
symbol:   variable MSG
location: class AloMundo
1 error
C:\Program Files\Java\jdk1.7.0_79\bin>
    
```

Talvez outro profissional seja um pouco mais experiente do que você e já tenha passado por essa situação.

O importante é você aprender, depurar e resolver o erro.

```
AloMundo.java - Bloco de notas
Arquivo  Editar  Formatar  Exibir  Ajuda
public class AloMundo {
    public static void main (String arg []){
        String msg = "Mundo";
        System.out.println("ola " + MSG);
    }
}
```

Em nosso exemplo, a única alteração realizada no código para gerar esse **1 error** foi na definição de uma variável denominada *msg*. Isso mesmo! Fizemos apenas uma alteração para maiúscula na variável *MSG*.

```
AloMundo.java - Bloco de notas
Arquivo  Editar  Formatar  Exibir  Ajuda
public class AloMundo {
    public static void main (String arg []){
        String msg = "Mundo";
        System.out.println("ola " + MSG);
    }
}
```

Veja onde está o erro. **Não** precisa alterar o seu código, pois nosso objetivo não é "aprofundar-nos em erros", mas em códigos perfeitos!

```
AloMundo.java - Bloco de notas
Arquivo  Editar  Formatar  Exibir  Ajuda
public class AloMundo {
    public static void main (String arg []){
        String msg = "Mundo";
        System.out.println("ola " + MSG);
    }
}
```

Em um primeiro momento, pode não estar claramente visível.

```
AloMundo.java - Bloco de notas
Arquivo  Editar  Formatar  Exibir  Ajuda
public class AloMundo {
    public static void main (String arg []){
        String msg = "Mundo";
        System.out.println("ola " + MSG);
    }
}
```

No entanto, o fato de definir uma variável msg utilizando caixa-baixa (minúscula, indicada na **caixa verde**) e, no decorrer do código, executar uma concatenação (junção) de **strings** (+) de uma frase "Ola", com uma variável MSG em "caixa-alta" (maiúscula, indicada na **caixa vermelha**) fará com que qualquer compilador indique uma variável não definida, não encontrada ou não declarada.

```
AloMundo.java - Bloco de notas
Arquivo  Editar  Formatar  Exibir  Ajuda
public class AloMundo {
    public static void main (String arg []){
        String msg = "Mundo";
        System.out.println("ola " + MSG);
    }
}
```

Isso irá ocorrer não apenas para a linguagem Java, que é sensível,
mas em qualquer codificação.

```
AloMundo.java - Bloco de notas
Arquivo  Editar  Formatar  Exibir  Ajuda
public class AloMundo {
    public static void main (String arg []){
        String msg = "Mundo";
        System.out.println("ola " + MSG);
    }
}
```

Só por isso? Sim, só por isso.

Atenção! Veremos as junções (concatenação) de string em um capítulo específico.


```
AloMundo.java - Bloco de notas
Arquivo  Editar  Formatar  Exibir  Ajuda
public class AloMundo {
    public static void main (String arg []){
        String msg = "Mundo";
        System.out.println("ola " + MSG);
    }
}
```

Uma variável *msg* (minúscula) é diferente de uma variável *MSG* (maiúscula). Isto é, palavras maiúsculas são diferentes de minúsculas.

```
AloMundo.java - Bloco de notas
Arquivo  Editar  Formatar  Exibir  Ajuda
public class AloMundo {
    public static void main (String arg []){
        String msg = "Mundo";
        System.out.println("ola " + MSG);
    }
}
```

Parece óbvio, mas o que aconteceu no código simulador de erro foi uma troca de minúscula por maiúscula.

Esse erro é comum entre os principiantes.

AloMundo.java - Bloco de notas

Arquivo Editar Formatar Exibir Ajuda

```
public class AloMundo {  
    public static void main (String arg []){  
        String msg = "Mundo";  
        System.out.println("ola " + MSG);  
    }  
}
```

Para corrigir o erro, basta reescrever *msg* com caixa-baixa.

```
AloMundo.java - Bloco de notas
Arquivo  Editar  Formatar  Exibir  Ajuda
public class AloMundo {
    public static void main (String arg []){
        String msg = "Mundo";
        System.out.println("ola " + msg);
    }
}
```

Com isso, voltamos ao código inicial, regravamos e recompilamos. Dessa forma, a tela estará sem erros novamente (linha do *prompt* vazia).