

## 1.1 Seu primeiro programa

Existem 3 etapas para o desenvolvimento em Java:

- Crie um programa em um arquivo chamado MyPrograma.java
- Compile-o no terminal digitando `javac MyPrograma.java`
- Execute-o no terminal digitando `java MyPrograma`

Termo código

↳ se refere ao texto do programa

Termo codificação

↳ se refere ao ato de criar e editar o código

Criando um programa

Um programa Java é uma sequência de caracteres, como um parágrafo ou poema, armazenado em um arquivo com extensão .java

compilando um programa

Um compilador é um aplicativo que traduz um programa da linguagem Java para uma linguagem mais adequada para execução. O compilador recebe um arquivo com extensão .java como entrada (seu programa) e produz um arquivo com o mesmo nome, mas com extensão .class (a versão do computador). Compilamos com o comando `javac` seguido do nome do arquivo.

executar um programa

Depois do compilador, o executor, a Máquina Virtual Java instrui o computador a seguir as instruções do seu programa, basta digitar o comando `java` seguido do nome do programa, no terminal.



## Programa 1.1.1 Olá, Mundo

```

// arquivo de texto (nome)
class public OláMundo { // método main()
    // declarações
    public static void main (String[] args) {
        // imprime "Olá mundo" no terminal.
        System.out.println("Olá, Mundo"); // corpo
    }
}

```

compilado e executado

- ↳ javac OláMundo.java
- ↳ java OláMundo
- = Olá, Mundo

Este programa tem uma única classe chamada `OláMundo` e possui um único método chamado `main()`.

- classe = programa
- programação = especificar um nome de classe e uma sequência de instruções para sua função `main()`.

O programa 1.1.1, possui duas instruções:

- A primeira é um comentário para documentar o programa, por um único linha e colocada '//'. O Java ignora os comentários na compilação.
- A segunda, é chamado o método `System.out.println()` para imprimir uma mensagem de texto.

Quando digitado `java` seguido do nome de uma classe, o sistema chama o método `main()` e executa as instruções uma por uma.

erros podem ser corrigido examinando o programa na criação

- Erros de compilação: impede o compilador de realizar a tradução.
- Erros de tempo de execução: o programa tenta realizar uma operação inválida.
- Erros lógicos: Executa e produz resposta errada, conhecido como bugs.



## Programa 1.1.2 Usando um argumento de linha de comando

```
classe public UseArgument {
```

```
    public static void main (String[] args {
```

```
        System.out.print ("Olá, ");
```

```
        System.out.print (args[0]);
```

```
        System.out.println (" Como vai você?");
```

### Compilado e executado

```
% javac UseArgument.java
```

```
' java UseArgument Alice
```

```
Olá, Alice, Tudo bem?
```

```
' java UseArgument Bob
```

```
Olá, Bob, Tudo bem?
```

Alice ← sequência de entrada



← caixa preta



Olá, Alice, Tudo bem? ← sequência de saída

### Entrada e saída

Sempre que executado o programa, ele aceita o argumento de linha de comando digitado após o nome do programa e o imprime de volta no terminal como parte do resultado.

Args[0] é o primeiro argumento de comando que é digitado após o nome do programa e args[1] é o segundo e assim por diante.

System.out.print(), imprime apenas o string especificado (e não um caractere de quebra de linha)

O programa UseArgument implementa uma função que mapeia uma sequência de caracteres como entrada (arquivo.java) e produz outra sequência de caracteres como saída (o arquivo class → mensagem exibida no terminal). Podemos pensar no programa java como uma caixa preta que converte a sequência de entrada em alguma sequência de saída.