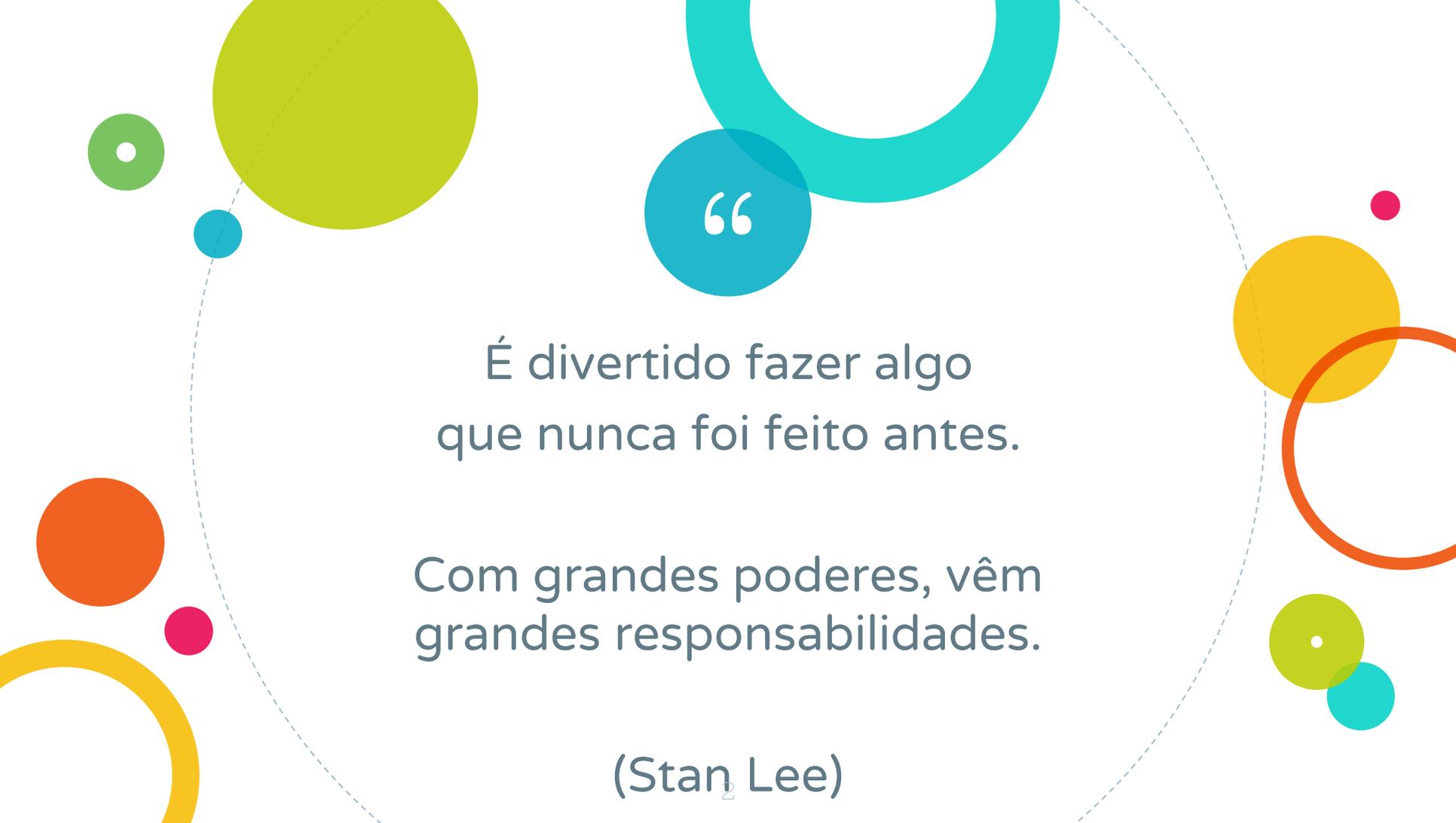


A decorative graphic featuring a large, light blue dashed circle that frames the central text. Various solid-colored circles in shades of teal, green, yellow, orange, and pink are scattered around the perimeter. Some circles are solid, while others are dashed outlines. A large teal ring is in the top left, a yellow ring is in the bottom right, and a large green circle is in the top right.

# Linguagem Java

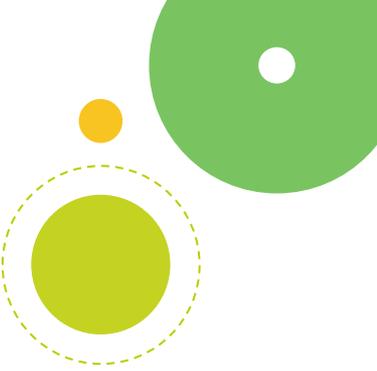
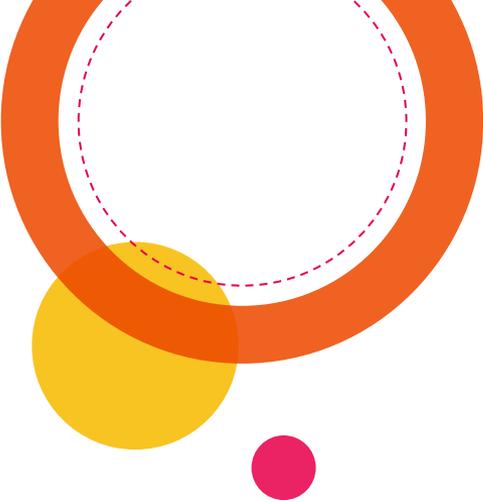
A decorative graphic featuring a large dashed white circle. Inside and around this circle are various colored shapes: a large green circle at the top left, a large cyan ring at the top center, a large yellow circle at the top right, a large orange circle at the bottom left, and a large yellow circle at the bottom right. Smaller circles in green, blue, orange, and cyan are scattered throughout. A large white quotation mark is centered at the top.

“

É divertido fazer algo  
que nunca foi feito antes.

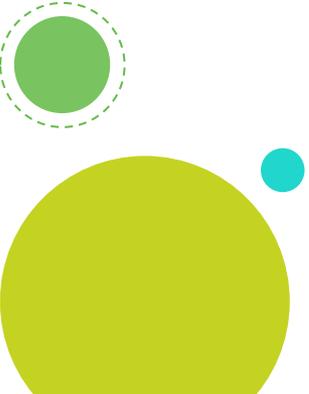
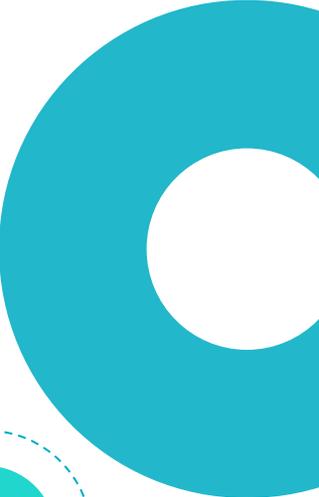
Com grandes poderes, vêm  
grandes responsabilidades.

(Stan Lee)



1

História



# Motivação

- ◎ Problemas nas linguagens na década de 1990
  - ◎ Gerenciamento de memória
  - ◎ Organização do código
  - ◎ Falta de bibliotecas
  - ◎ Reescrita de código para diferentes Sistemas Operacionais (SO)

# Propósitos

- ⦿ Uso em pequenos dispositivos (TV, videocassetes, aspiradores...)
- ⦿ Lançamento focado no uso em clientes web (browsers ou navegadores)
- ⦿ Rodar pequenas aplicações (applets)



# Linha do Tempo

- © 1992: Sun criou um time (Green Team) para desenvolver inovações tecnológicas, liderada por James Gosling (pai do Java)
- © Projeto \*7 (star seven) – “Oak” e Duke
- © 1995: Oak para Java
- © 2004: mais de 3 milhões de desenvolvedores
- © 2006: GNU General Public License (GPL)
- © 2008: Oracle Corporation adquire a Sun Microsystem (US\$ 7,4 bilhões)



The background features several decorative elements: a large orange ring with a dashed red inner circle in the top left; a large teal ring in the bottom right; a large green circle with a white center in the top right; a large yellow circle in the middle left; a large green circle with a dashed green border in the middle right; a large teal circle with a dashed teal border in the bottom right; a large green circle with a dashed green border in the bottom left; a large yellow circle in the middle left; a small pink circle in the middle left; a small orange circle in the top right; a small teal circle in the bottom right; and a small cyan circle in the bottom left.

2

Características

# Vantagens

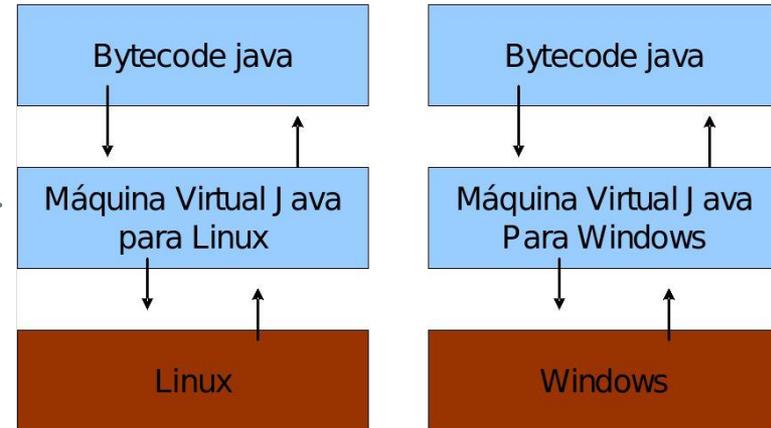
- ◎ Orientada a objetos
- ◎ Portabilidade “escreva uma vez, execute em qualquer lugar” (*write onde, run anywhere*)
- ◎ Recursos de rede
- ◎ Segurança
- ◎ Sintaxe similar a linguagem C
- ◎ Facilidades de internacionalização
- ◎ Vasto conjunto de API's (Application Programming Interface – Interface de Programação de Aplicação)
- ◎ Facilidades para criação de programas distribuídos e multitarefas
- ◎ “*Garbage collector*”

# Paradigmas de Programação

- © Está diretamente relacionado à **forma de pensar** do programador e como ele busca **solucionar os problemas**.
- © **Paradigma Estruturado** (ou Imperativo) – Procura uma forma de quebrar os problemas. Qualquer problema pode ser resolvido com 3 estruturas: Sequencial, Condicional e Repetição. Analisa o problema tentando relacionar as ações que deverão ser executadas e como poderão ser divididas em módulos.
- © **Paradigma Orientado a Objetos** – O problema é tratado como uma coleção de objetos interagindo por meio de trocas de mensagem. Os **objetos** são estruturas que contém atributos e ações. Um conjunto de objetos com o mesmo comportamento dá origem a uma **classe**. Analisa o problema identificando os objetos que compõem o problema e como interagem

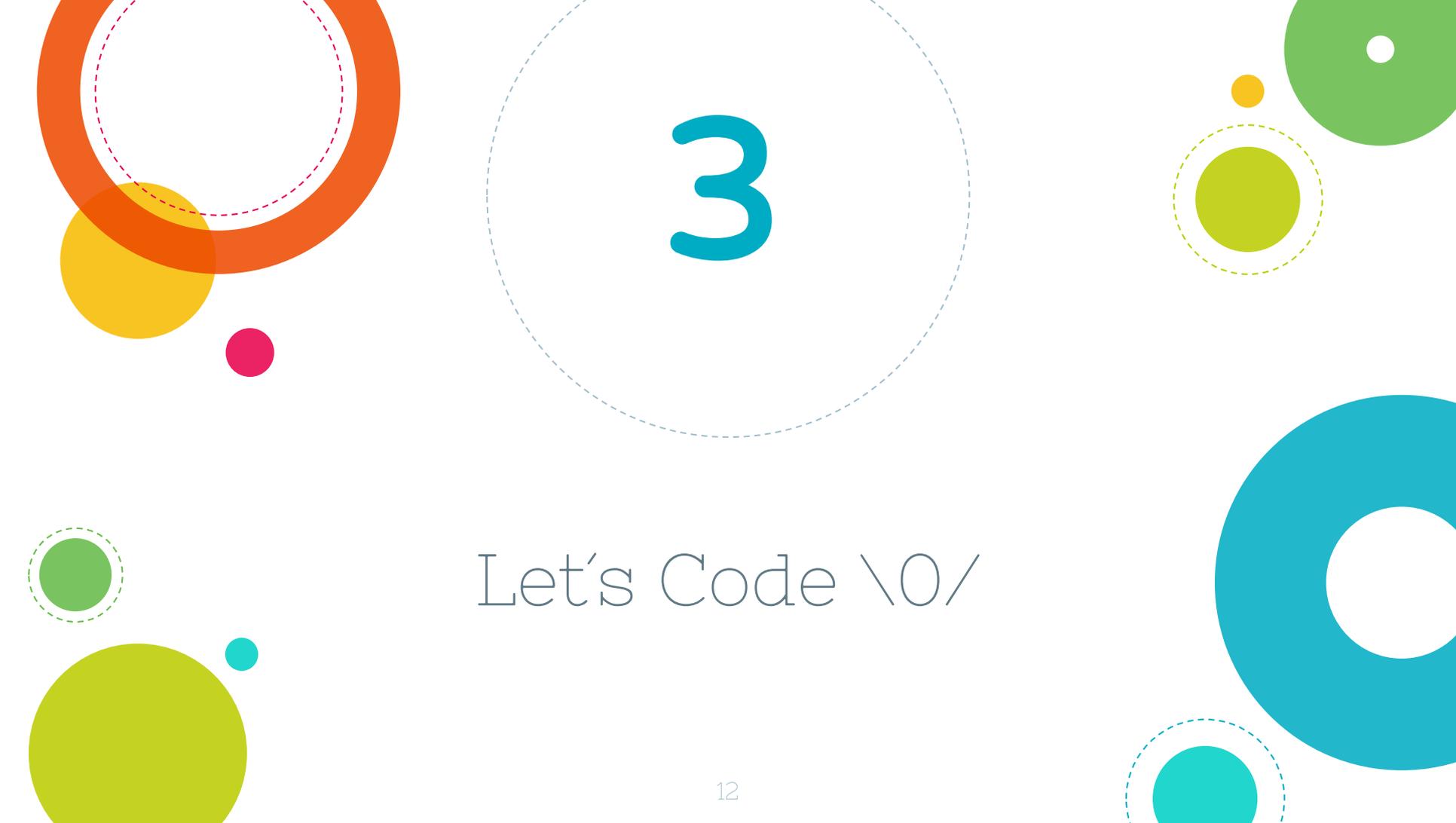
# Máquina Virtual Java (JVM)

- © É um programa que age como um computador completo, com todas as instâncias de um sistema operacional funcional
- © Camada de isolamento que conversa com qualquer tecnologia
- © Interpreta o código de máquina (“bytecode”) gerado pelo compilador



# Componentes

- ◎ JVM: Vem sempre acompanhada
- ◎ JRE: Java Runtime Environment. Ambiente de execução java. Bibliotecas + JVM
- ◎ JDK: Java Development Kit. Utilizamos o Java SE (Standard Edition). JRE + Compilador
- ◎ Última Versão: Java SE 17
- ◎ <https://www.oracle.com/br/java/technologies/javase-downloads.html>

The background features several decorative elements: a large orange ring with a dashed red inner circle in the top left; a large teal ring with a white center in the bottom right; a large green circle with a white center in the top right; a large yellow circle in the middle left; a large green circle with a dashed green border in the bottom left; and a large teal circle with a dashed teal border in the bottom right. There are also smaller solid circles in pink, cyan, and orange scattered throughout.

3

Let's Code \0/

# Código Fonte

- Utilize qualquer editor de texto. De preferência um que tenha *syntax highlighting* (Notepad++)

```
1 public class Ex1 {
2     public static void main(String[] args) {
3         System.out.println("EXPERIMENTANDO a Linguagem JaVa ");
4         System.out.print("ADS - PUC Go ");
5         System.out.println("***** ***** ");
6         System.out.print("ADS \n PUC Go ");
7     }
8 }
```

- Salve o arquivo com o mesmo nome do programa (na linha 1) com a extensão `.java` Nesse exemplo, **Ex1.java**
- Obs1: Java é “Case sensitive”, ou seja: diferencia letras maiúsculas e minúsculas.
- Obs2: O nome do arquivo deve ser o mesmo da “classe” (linha1)

# Prompt de Comando

- ⦿ É um interpretador de linha de comando
  - ⦿ Windows: **CMD** (digite na barra de pesquisa)
  - ⦿ Linux: **Monitor**: Está dentro das ferramentas de desenvolvimento
- ⦿ Prompt (aguardando comandos)
  - ⦿ Windows: caractere “Sinal de Maior”: **>**
  - ⦿ Linux: caractere “Cifrão”: **\$**

```
CA Prompt de Comando
Microsoft Windows [versão 10.0.19042.1165]
(c) Microsoft Corporation. Todos os direitos reservados.

C:\Users\lucil>
```

# Comandos do Prompt

- ⦿ Limpa a tela: **cls** (Windows) ou **clear** (Linux)
- ⦿ Mostra o conteúdo da pasta: **dir** (Windows) ou **ls** (Linux)
- ⦿ Percorrer os diretórios: **CD caminho**

```
C:\>CD Users

C:\Users>CD lucil\eclipse-workspace\CMP1046\src

C:\Users\lucil\eclipse-workspace\CMP1046\src>DIR
O volume na unidade C é OS
O Número de Série do Volume é B2B8-48DD

Pasta de C:\Users\lucil\eclipse-workspace\CMP1046\src

12/08/2021  20:13    <DIR>          .
12/08/2021  20:13    <DIR>          ..
12/08/2021  20:14                442 Ex1.class
12/08/2021  20:13                224 Ex1.java
                2 arquivo(s)                666 bytes
                2 pasta(s)  37.911.416.832 bytes disponíveis

C:\Users\lucil\eclipse-workspace\CMP1046\src>CD..

C:\Users\lucil\eclipse-workspace\CMP1046>CD\

C:\>
```

# Compilação

- ◎ O compilador da linguagem Java é o **javac**
- ◎ Chame o compilador e digite o nome do arquivo a ser compilado:

**javac Ex1.java**

```
C:\Code>dir
O volume na unidade C é OS
O Número de Série do Volume é B2B8-48DD

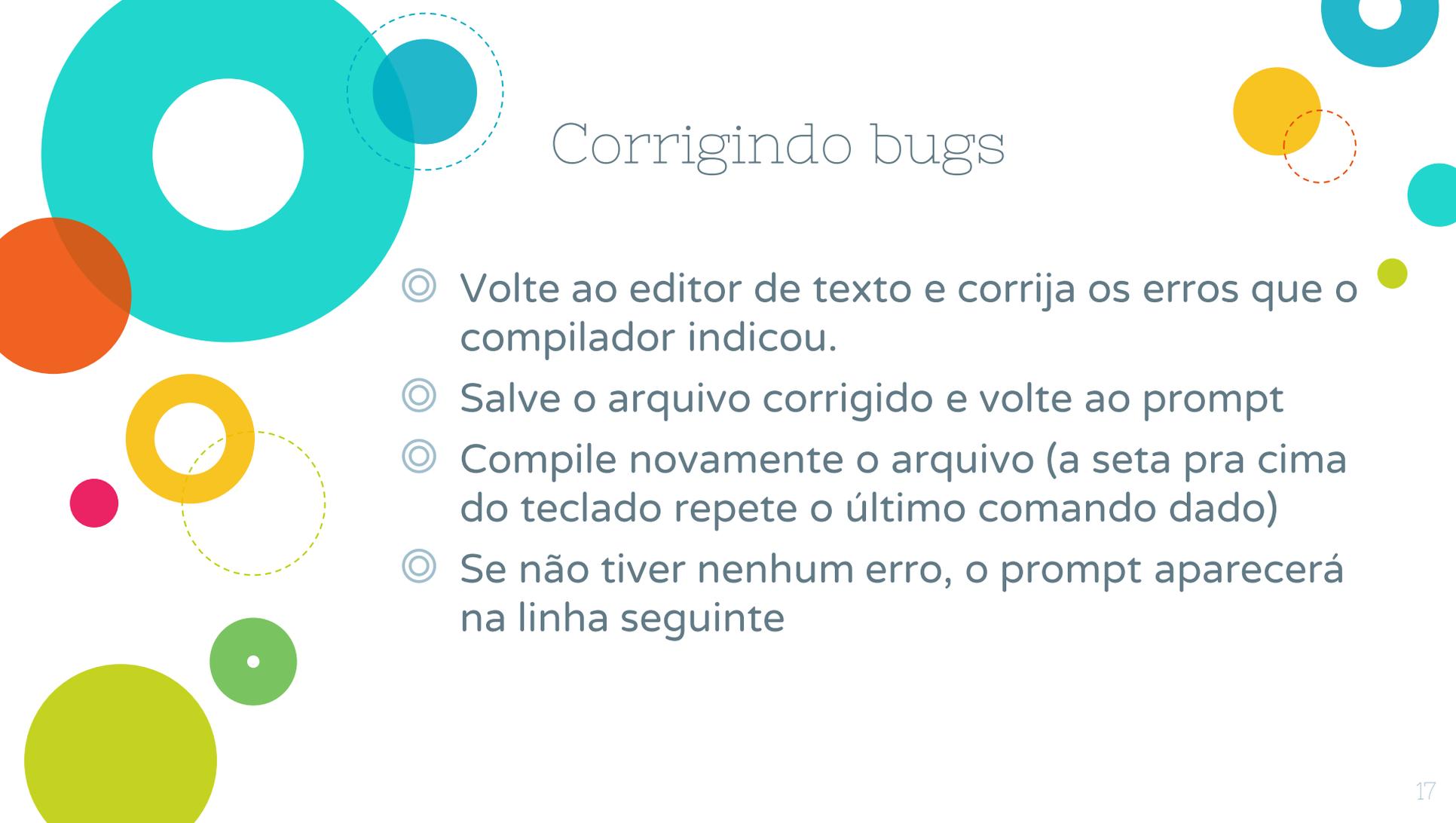
Pasta de C:\Code

13/08/2021  21:07    <DIR>          .
13/08/2021  21:07    <DIR>          ..
13/08/2021  21:02                265 Ex1.java
                1 arquivo(s)          265 bytes
                2 pasta(s) 37.945.765.888 bytes disponíveis

C:\Code>javac Ex1.java
Ex1.java:5: error: unclosed string literal
    System.out.println("***** ***** ");
                       ^
1 error

C:\Code>
```

- ◎ Se o código fonte tiver algum erro, o compilador apresentará uma mensagem, indicando a linha e o erro.



## Corrigindo bugs

- ⦿ Volte ao editor de texto e corrija os erros que o compilador indicou.
- ⦿ Salve o arquivo corrigido e volte ao prompt
- ⦿ Compile novamente o arquivo (a seta pra cima do teclado repete o último comando dado)
- ⦿ Se não tiver nenhum erro, o prompt aparecerá na linha seguinte

# Executando o código

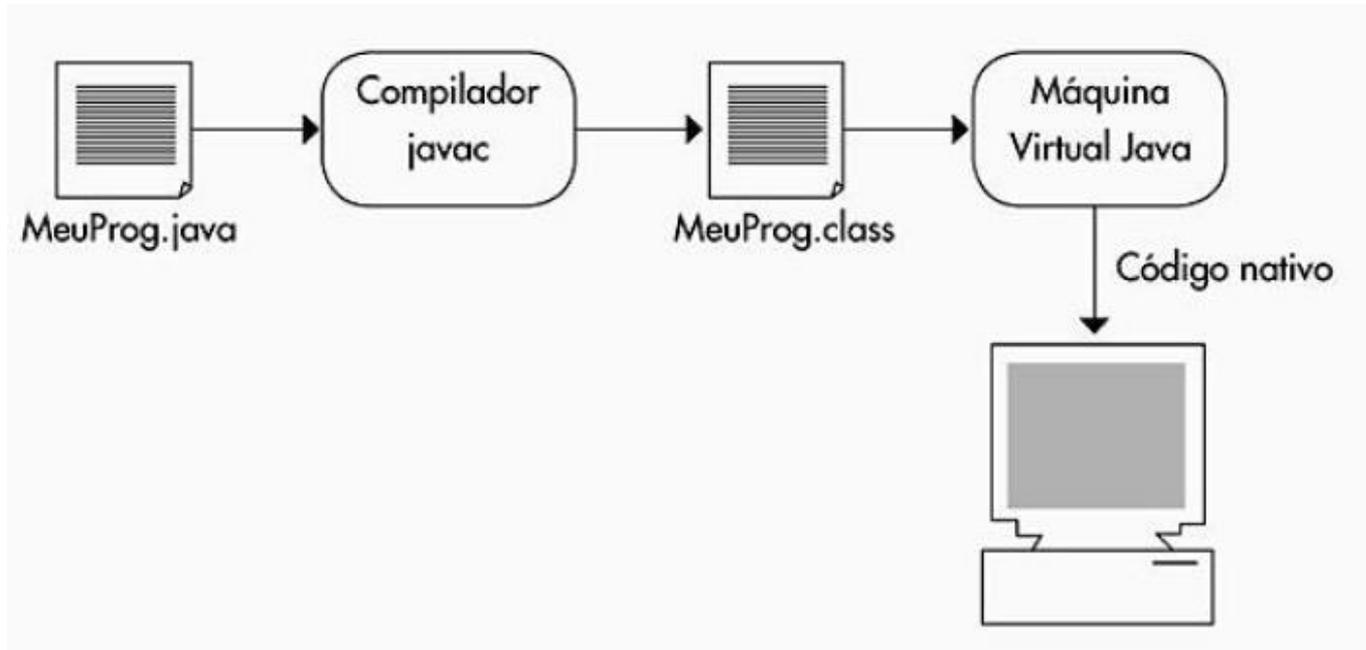
- © Para executar o código, chame o java seguido do nome do arquivo (sem extensão): **java Ex1**

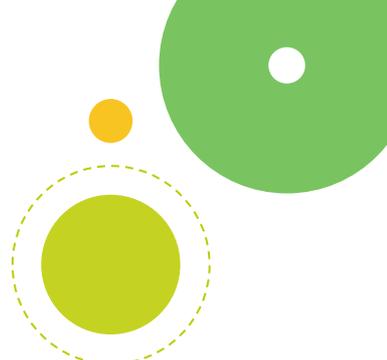
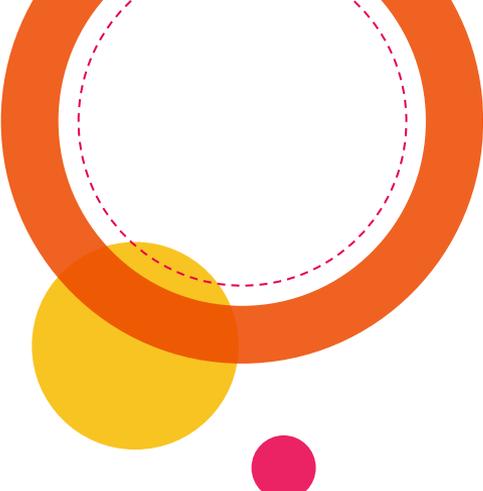
 Prompt de Comando

```
C:\Code>javac Ex1.java

C:\Code>java Ex1
EXPERIMENTANDO a Linguagem JaVa
ADS - PUC Go *****
ADS
  PUC Go
C:\Code>
```

# Processo de execução





4

# Exercícios





## Exercício 3.2

- © Faça um programa que imprima um menu semelhante a este:

```
***** STAR PLAY *****
```

```
1 — Iniciar Jogo
```

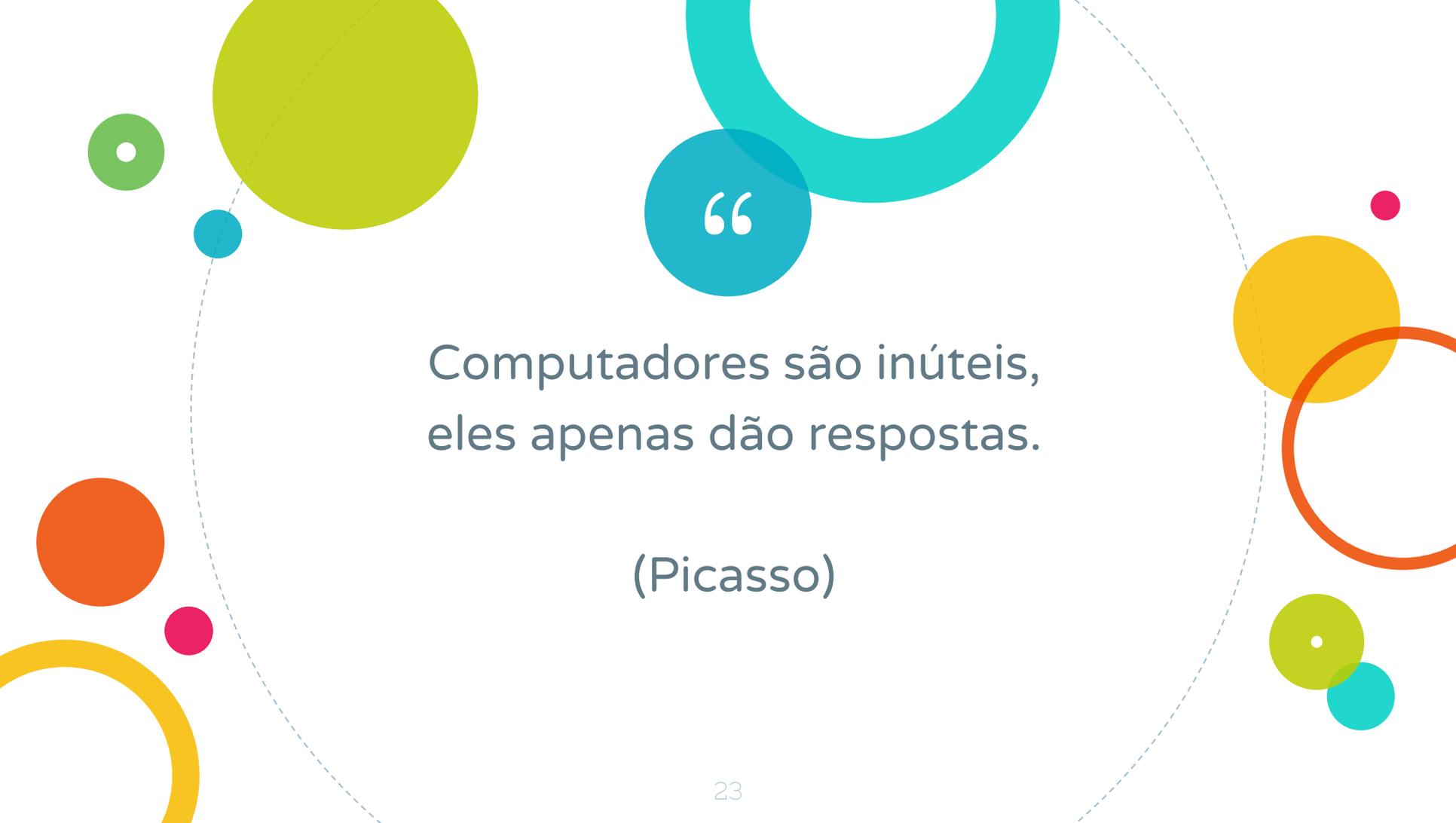
```
2 — Restaurar último Jogo
```

```
3 — Configurar Jogo
```

```
0 — Sair
```

```
*****
```

```
OPÇÃO →
```

A decorative graphic consisting of various colored circles (green, blue, orange, yellow, pink) and a dashed light blue line that forms a partial circle around the central text. A large teal ring is positioned at the top center.

“

Computadores são inúteis,  
eles apenas dão respostas.

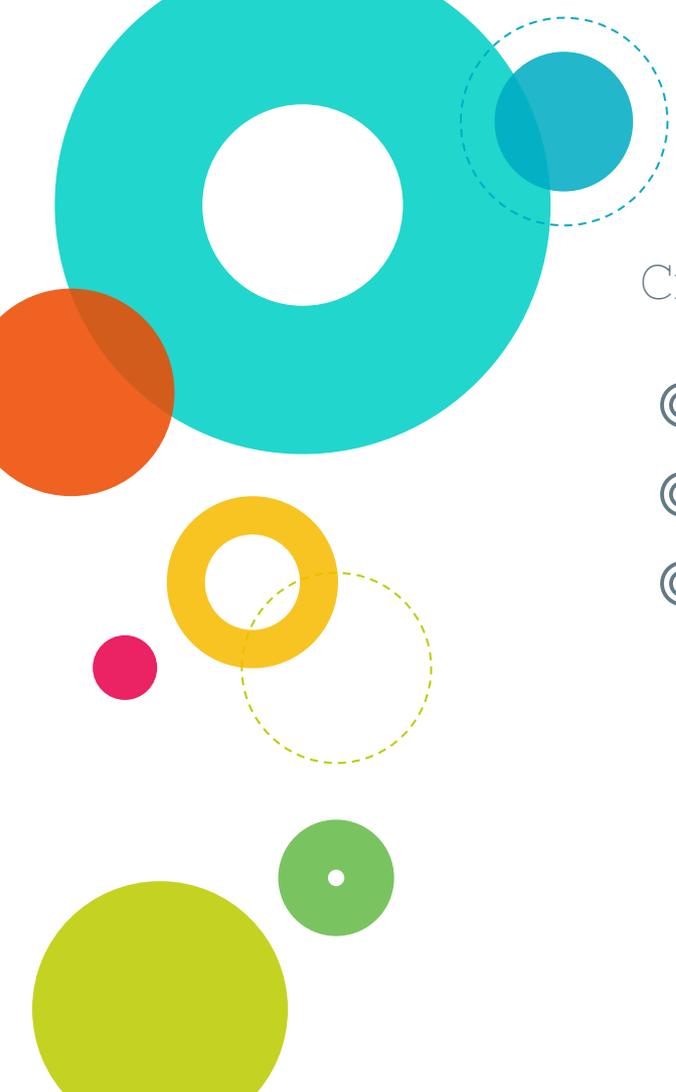
(Picasso)

Obrigada!



Perguntas?

[professora@lucilia.com.br](mailto:professora@lucilia.com.br)



## Créditos

© Template: SlidesCarnival

© [www.guj.com.br](http://www.guj.com.br)

© Fundamentos da Programação