

# Fundamentos de Desenvolvimento em Java

## Escrever programas em Java usando seus elementos básicos

Professor Vitor Amadeu Souza

[vitor.asouza@prof.infnet.edu.br](mailto:vitor.asouza@prof.infnet.edu.br)



# Referências

1. Java: A Beginner's Guide, 9th Edition.
2. Head First Java, 3rd. Edition
3. Learning Java, 6th Edition
4. Java in a Nutshell, 8th Edition
5. Sololearn Java e Sololearn Intermediate
6. Core Java for the Impatient, 3rd Edition (exercícios)

# Histórico

- \* Criado em 1991 por James Gosling da Sun Microsystems.
- \* Inicialmente chamada OAK em homenagem à um Carvalho em frente a janela de Gosling
- \* Seu nome foi mudado para Java devido a existência de uma linguagem com o nome OAK

# Histórico

- \* A motivação original era a necessidade de uma linguagem independente de plataforma que podia ser utilizada em vários produtos eletrônicos, tais como torradeiras e refrigeradores.
- \* Ao mesmo tempo, a World Wide Web e a Internet foram ganhando popularidade. Gosling achava que a linguagem Java poderia ser usada para programação da Internet

# Histórico



# Histórico

- \* A motivação original era a necessidade de uma linguagem independente de plataforma que podia ser utilizada em vários produtos eletrônicos, tais como torradeiras e refrigeradores.
- \* Ao mesmo tempo, a World Wide Web e a Internet foram ganhando popularidade. Gosling achava que a linguagem Java poderia ser usada para programação da Internet

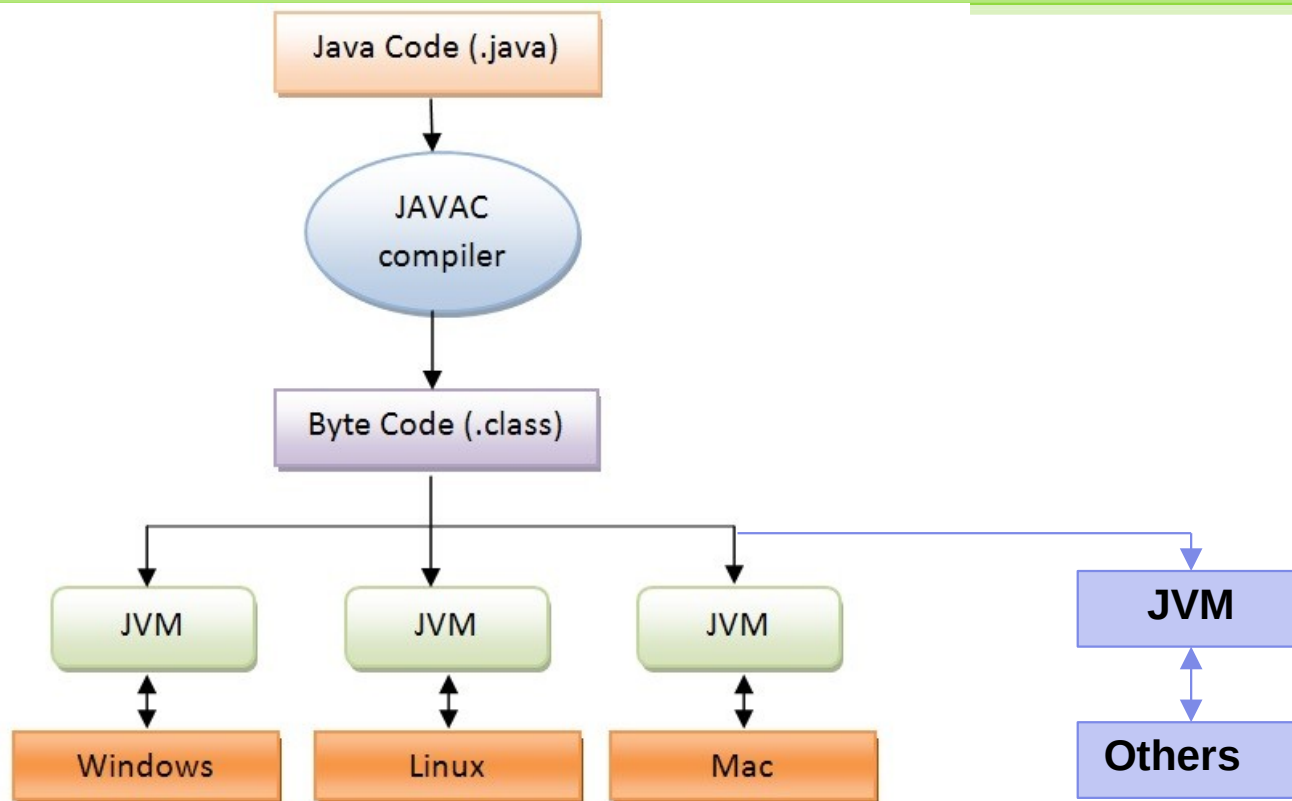
# Histórico – James Gosling



[https://pt.wikipedia.org/wiki/James\\_Gosling](https://pt.wikipedia.org/wiki/James_Gosling)



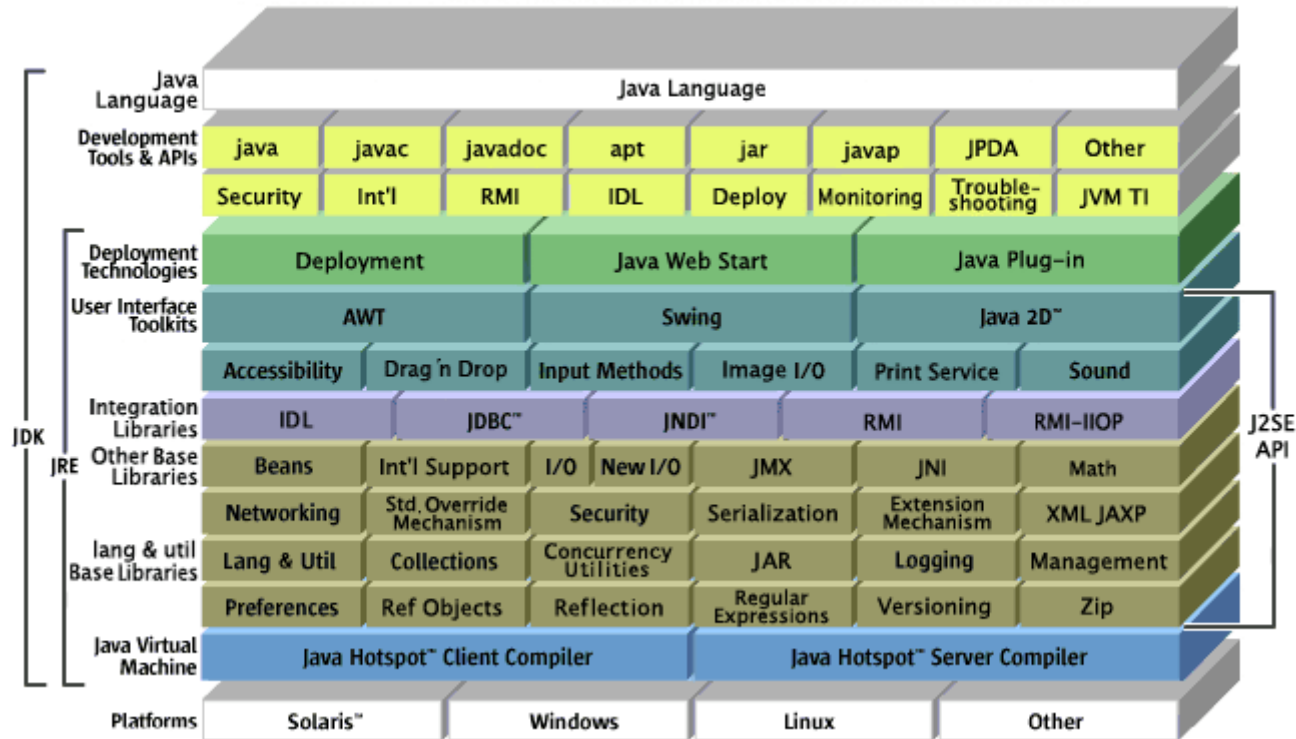
# Máquina virtual (JVM – Java Virtual Machine)





# Tecnologia Java

## Java™ 2 Platform Standard Edition 5.0



# Tecnologia Java

Compilador → javac.exe

Interpretador → java.exe

Gerador de documentos → javadoc.exe

Empacotamento de classes e arquivos → jar.exe

# Oracle x Sun



/tecnologia / fusões e aquisições

20/04/09 - 09h55 - Atualizado em 20/04/09 - 13h23

## Oracle anuncia compra da Sun por mais de US\$ 7 bilhões

Pelo acordo, cada ação está avaliada em US\$ 9,5.  
Negociações entre IBM e Sun fracassaram no início do mês.

Do G1, com agências

clique para ampliar



A Oracle anunciou nesta segunda-feira (20) que comprará a Sun Microsystems por US\$ 9,5 por ação, numa operação que avalia a fabricante de servidores de alta potência em mais de US\$ 7

celular rss

ACORDO ORTOGRÁFICO

saiba mais



/ plantão



SEX, 7/1/2011

13h33 | amazonia

Índios suruí apostam no mercado de carbono para conservar sua terra em RO

TER, 4/1/2011

20h08 | amazonia

Filhote de peixe-boi sem a mãe é resgatado no Amazonas

12h48 | amazonia

Expedição faz levantamento inédito do Parque da Serra do Pardo, no



ps://g1.globo.com/Noticias/Vestibular/0,,MUL936903-5604,00-IMPRIMA+GUIA+DA+REFORMA+ORTOGRAFICA.html

<https://g1.globo.com/Noticias/Tecnologia/0,,MUL1091457-6174,00-ORACLE+ANUNCIA+COMPRA+DA+SUN+POR+MAIS+DE+US+BILHOES.html>

# Instalação do Apache NetBeans 18



The screenshot shows the Apache NetBeans website. At the top, there's a navigation bar with links: Community, Participate, Blog, Get Help, Plugins, and Download. Below this, a large blue banner features the text "Latest release" and "Apache NetBeans 18" with a green "Download" button. Underneath the banner is the Apache NetBeans logo, a 3D cube with green, yellow, and blue faces, followed by the text "Apache NetBeans" and "Fits the Pieces Together". Below this, a tagline reads "Development Environment, Tooling Platform and Application Framework." At the bottom, there are three colored boxes: a blue box with "Fast & Smart Editing" and "Apache NetBeans is much", a green box with "Java, JavaScript, PHP, HTML5, CSS, and More", and a red box with "Cross Platform" and "Apache NetBeans can be installed on all operating".

Apache NetBeans

Community Participate Blog Get Help Plugins Download

Latest release

Apache NetBeans 18

Download

Apache NetBeans

Fits the Pieces Together

Development Environment, Tooling Platform and Application Framework.

Fast & Smart Editing  
Apache NetBeans is much

Java, JavaScript, PHP, HTML5, CSS, and More

Cross Platform  
Apache NetBeans can be installed on all operating

<https://netbeans.apache.org/>

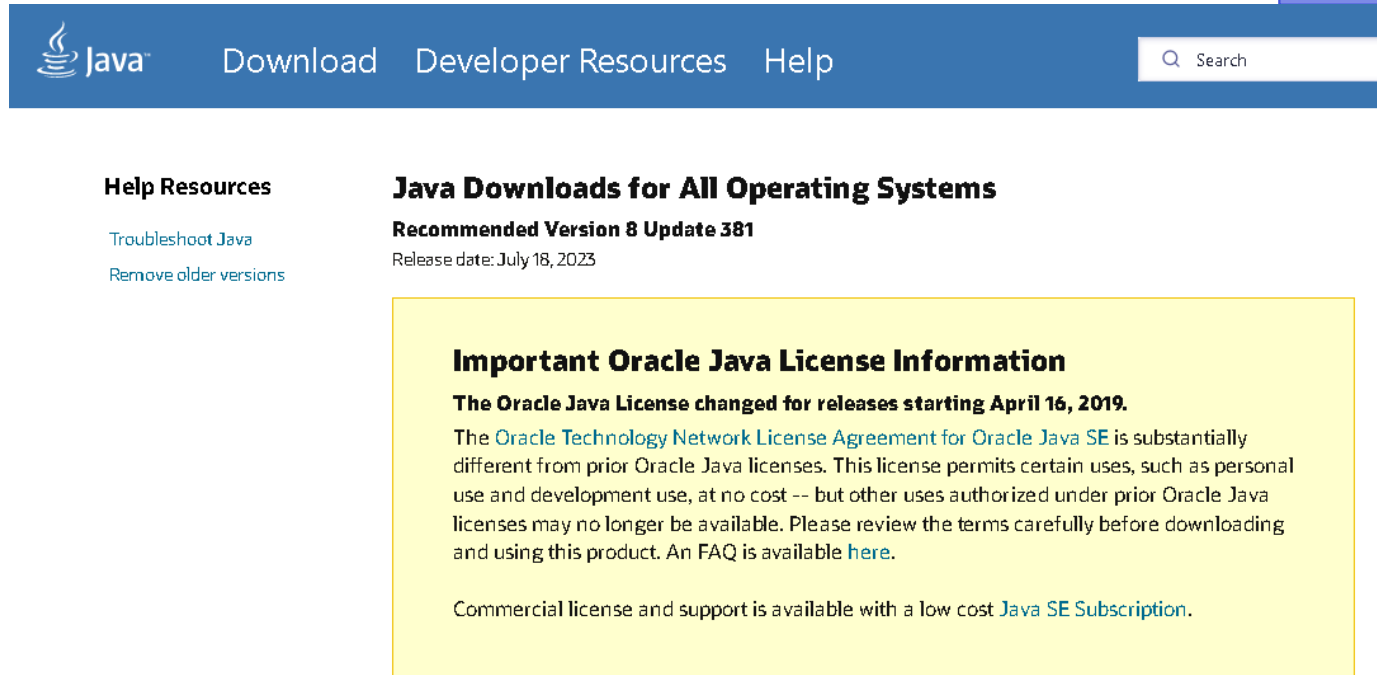


# JRE (Java Runtime Environment)

O Java Runtime Environment (JRE) é um software necessário para que os programas em Java funcionem corretamente. O Java é uma linguagem de programação que viabiliza muitas aplicações móveis e Web usadas hoje em dia. O JRE é a tecnologia subjacente que estabelece a comunicação entre o programa em Java e o sistema operacional. Ele atua como um tradutor e facilitador, fornecendo todos os recursos para que o software que você cria usando a linguagem funcione em qualquer sistema operacional sem modificações adicionais.

Fonte: <https://aws.amazon.com/pt/what-is/java-runtime-environment/>

# JRE (Java Runtime Environment)



The screenshot shows the top navigation bar of the Java website with the Java logo, links for Download, Developer Resources, and Help, and a search bar. Below the navigation bar, there are two main sections. On the left, under 'Help Resources', are links for 'Troubleshoot Java' and 'Remove older versions'. On the right, under 'Java Downloads for All Operating Systems', is the 'Recommended Version 8 Update 381' with a release date of July 18, 2023. Below this is a yellow box containing 'Important Oracle Java License Information', which states that the Oracle Java License changed for releases starting April 16, 2019, and provides details about the Oracle Technology Network License Agreement for Oracle Java SE. It also mentions that commercial license and support is available with a low cost Java SE Subscription.

**Help Resources**

- [Troubleshoot Java](#)
- [Remove older versions](#)

**Java Downloads for All Operating Systems**

**Recommended Version 8 Update 381**  
Release date: July 18, 2023

**Important Oracle Java License Information**

**The Oracle Java License changed for releases starting April 16, 2019.**

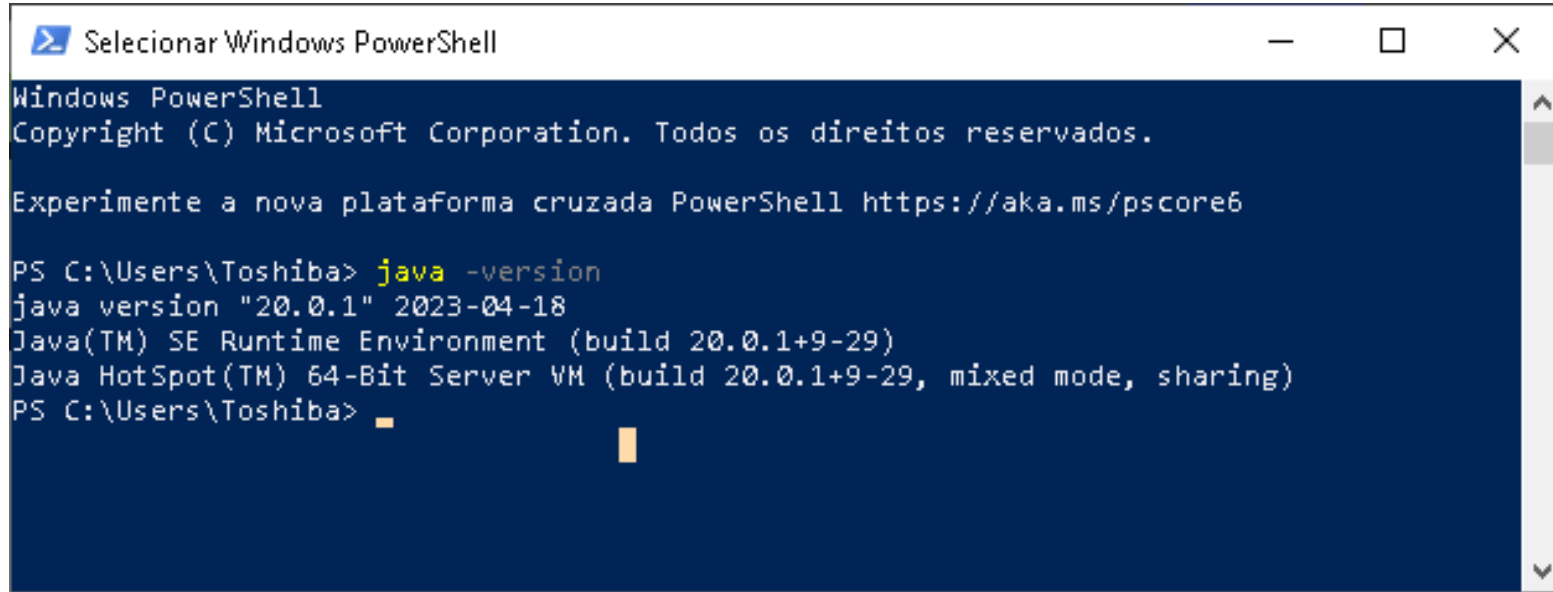
The [Oracle Technology Network License Agreement for Oracle Java SE](#) is substantially different from prior Oracle Java licenses. This license permits certain uses, such as personal use and development use, at no cost -- but other uses authorized under prior Oracle Java licenses may no longer be available. Please review the terms carefully before downloading and using this product. An FAQ is available [here](#).

Commercial license and support is available with a low cost [Java SE Subscription](#).

<https://www.java.com/en/download/manual.jsp>

# JRE (Java Runtime Environment)

java -version



```
Selecionar Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos os direitos reservados.

Experimente a nova plataforma cruzada PowerShell https://aka.ms/pscore6

PS C:\Users\Toshiba> java -version
java version "20.0.1" 2023-04-18
Java(TM) SE Runtime Environment (build 20.0.1+9-29)
Java HotSpot(TM) 64-Bit Server VM (build 20.0.1+9-29, mixed mode, sharing)
PS C:\Users\Toshiba>
```

# JDK (Java Runtime Environment)

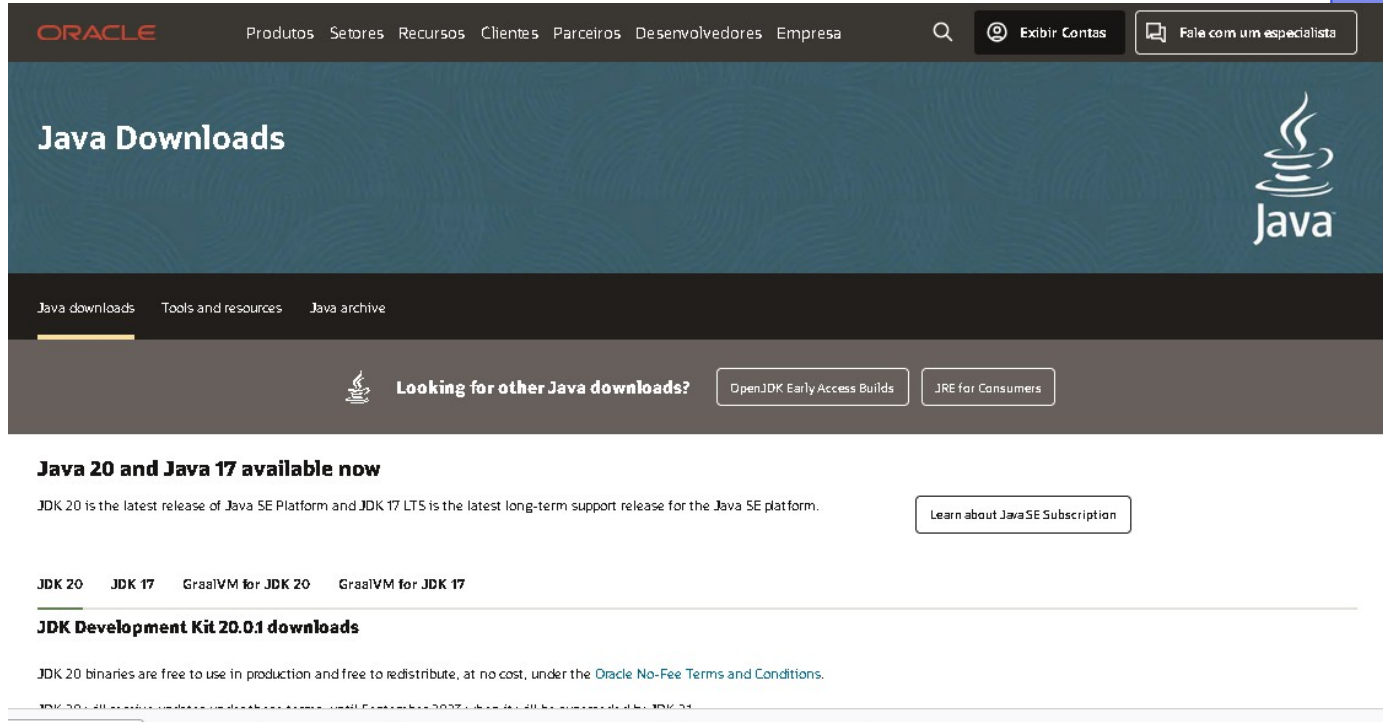
O JDK é Kit que provê ferramentas para o desenvolvimento de programas Java. Ou seja, ele contém um compilador, um depurador e o próprio JRE para você executar os seus programas.

É o JDK que nos provê o programa javac.exe, que compila códigos .java em bytecodes.class.

<https://dicasdeprogramacao.com.br/qual-a-diferenca-entre-jdk-jre-e-jvm/>



# JDK (Java Development Kit)

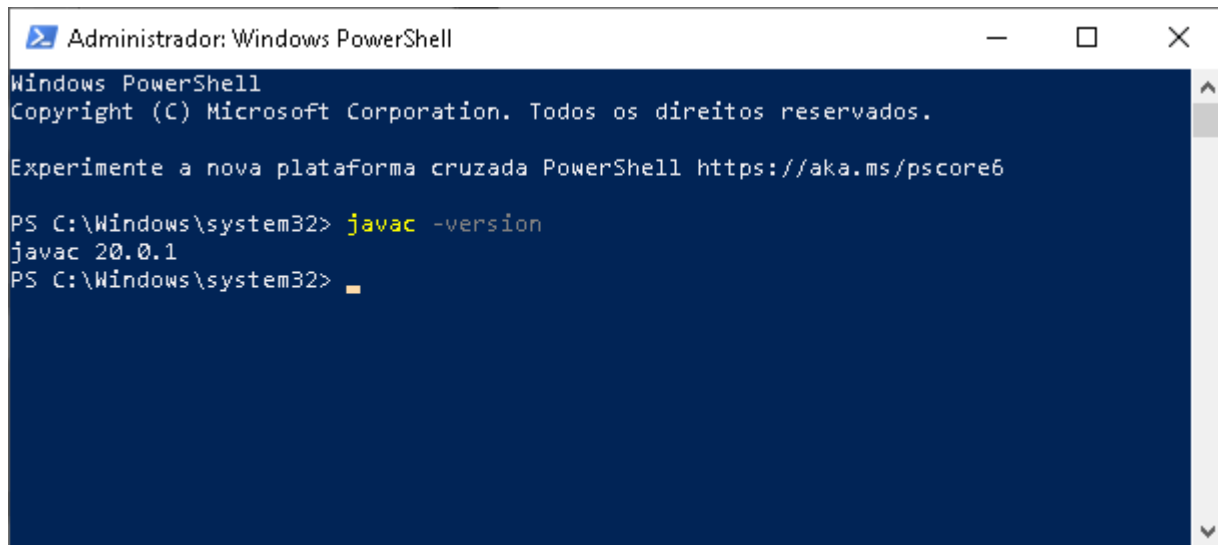


The screenshot shows the Oracle Java Downloads page. At the top, the Oracle logo is on the left, and navigation links for 'Produtos', 'Setores', 'Recursos', 'Clientes', 'Parceiros', 'Desenvolvedores', and 'Empresa' are in the center. On the right, there are links for 'Exibir Contas' and 'Fale com um especialista'. Below the navigation bar, the main heading is 'Java Downloads' with the Java logo to its right. A secondary navigation bar includes 'Java downloads' (highlighted), 'Tools and resources', and 'Java archive'. Below this, there's a section titled 'Looking for other Java downloads?' with buttons for 'OpenJDK Early Access Builds' and 'JRE for Consumers'. The main content area starts with 'Java 20 and Java 17 available now', followed by a paragraph stating 'JDK 20 is the latest release of Java SE Platform and JDK 17 LTS is the latest long-term support release for the Java SE platform.' and a 'Learn about Java SE Subscription' button. Below this is a navigation bar with links for 'JDK 20', 'JDK 17', 'GraalVM for JDK 20', and 'GraalVM for JDK 17'. The section title 'JDK Development Kit 20.0.1 downloads' is followed by a paragraph stating 'JDK 20 binaries are free to use in production and free to redistribute, at no cost, under the Oracle No-Fee Terms and Conditions.'

<https://www.oracle.com/br/java/technologies/downloads/>

# JDK (Java Development Kit)

`javac -version`



```
Administrador: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos os direitos reservados.

Experimente a nova plataforma cruzada PowerShell https://aka.ms/pscore6

PS C:\Windows\system32> javac -version
javac 20.0.1
PS C:\Windows\system32> 
```

# Há opções on-line...

**Online Java Compiler IDE**  
For Multiple Files, Custom Library and File Read/Write, use our new - [Advanced Java IDE](#)

```
1 public class MyClass {  
2     public static void main(String args[]) {  
3         int x=10;  
4         int y=25;  
5         int z=x+y;  
6  
7         System.out.println("Sum of x+y = " + z);  
8     }  
9 }
```

Execute Mode, Version, Inputs & Arguments

JDK 17.0.1  ☐ Interactive Stdin Inputs

CommandLine Arguments

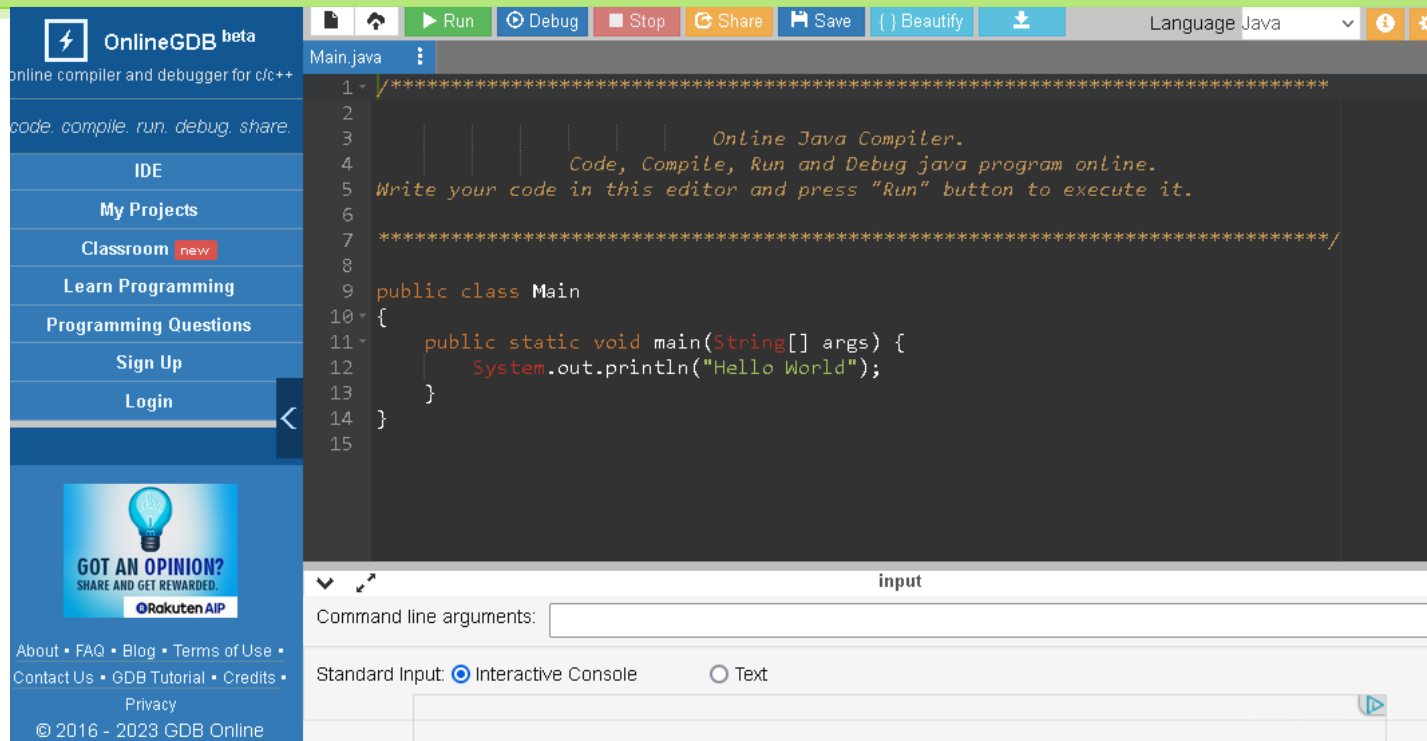
Result

CPU Time: 0.12 sec(s), Memory: 33476 kilobyte(s) compiled and executed in 0.781 sec(s)

Sum of x+y = 35

<https://www.jdoodle.com/online-java-compiler/>

# Há opções on-line...



The screenshot displays the OnlineGDB website interface. On the left is a blue sidebar with navigation links: "IDE", "My Projects", "Classroom" (with a "new" badge), "Learn Programming", "Programming Questions", "Sign Up", and "Login". Below these is a "GOT AN OPINION?" banner with a lightbulb icon and the Rakuten AIP logo. At the bottom of the sidebar are links for "About", "FAQ", "Blog", "Terms of Use", "Contact Us", "GDB Tutorial", "Credits", and "Privacy", along with the copyright notice "© 2016 - 2023 GDB Online". The main area features a top toolbar with buttons for "Run", "Debug", "Stop", "Share", "Save", "Beautify", and a "Language" dropdown set to "Java". Below the toolbar is a code editor with a dark background, showing a Java "Hello World" program. The code includes a multi-line comment at the top and a `public class Main` with a `main` method that prints "Hello World". At the bottom of the editor is an "input" section with a "Command line arguments:" field and "Standard Input:" options for "Interactive Console" (selected) and "Text".

```
1  /*****  
2  
3      Online Java Compiler.  
4      Code, Compile, Run and Debug java program online.  
5      Write your code in this editor and press "Run" button to execute it.  
6  
7  *****/  
8  
9  public class Main  
10 {  
11     public static void main(String[] args) {  
12         System.out.println("Hello World");  
13     }  
14 }  
15
```

[https://www.onlinegdb.com/online\\_java\\_compiler](https://www.onlinegdb.com/online_java_compiler)

# Primeiro programa em Java

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        System.out.println("Hello World");
    }
}
```

# Primeiro programa em Java

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        System.out.println("Hello World");
    }
}
```

→ \* Nome da classe, neste caso  
JavaApplication1

\* Todo o código em Java é escrito dentro de  
uma classe, observe a palavra-chave class

\* O identificador de acesso public indica que  
esta classe é acessível a outras classes de  
diferentes pacotes (coleções de classes)

# Primeiro programa em Java

```
package javaapplication1;
```

```
public class JavaApplication1
```

```
{  
    public static void main(String[] args)  
    {  
        System.out.println("Hello World");  
    }  
}
```

→ Início do bloco de instruções

# Primeiro programa em Java

```
package javaapplication1;  
  
public class JavaApplication1  
{  
    public static void main(String[] args)  
    {  
        System.out.println("Hello World");  
    }  
}
```

Nome do método principal (main) do programa. É a única função necessária para o funcionamento do Java



# Primeiro programa em Java

```
package javaapplication1;  
  
public class JavaApplication1  
{  
    public static void main(String[] args)  
    {  
        System.out.println("Hello World");  
    }  
}
```

**static (estático)** indica que o método main é compartilhado por todos os objetos criados a partir desta classe

# Primeiro programa em Java

```
package javaapplication1;  
  
public class JavaApplication1  
{  
    public static void main(String[] args)  
    {  
        System.out.println("Hello World");  
    }  
}
```

void indica o retorno da função, neste caso  
vazio

# Primeiro programa em Java

```
package javaapplication1;  
  
public class JavaApplication1  
{  
    public static void main(String[] args)  
    {  
        System.out.println("Hello World");  
    }  
}
```

É o argumento de entrada ao chamar o programa, um vetor de strings

# Primeiro programa em Java

```
package javaapplication1;  
  
public class JavaApplication1  
{  
    public static void main(String[] args)  
    {  
        System.out.println("Hello World");  
    }  
}
```

Escreve do padrão de saída o texto entre  
parêntesis

# Primeiro programa em Java

```
package javaapplication1;  
  
public class JavaApplication1  
{  
    public static void main(String[] args)  
    {  
        System.out.println("Hello World");  
    }  
}
```

Fim do método

Fim da classe



# Comentários

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        System.out.println("Hello World"); //Este é um comentário de linha
    }
}
```

# Comentários

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        System.out.println("Hello World");
        /*Este é um
           comentário de
           múltiplas linhas*/
    }
}
```

# Classe String

- \* String é uma classe que permite declarar objetos para armazenar uma sequência de caracteres.
- \* Dentro dela há inúmeros métodos para tratamento de texto.



# Classe String

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        String dado="Hello World";
        System.out.println(dado);
    }
}
```

# Classe String

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        String dado="Hello World";
        System.out.println(dado);
        System.out.println(dado.toUpperCase());
        System.out.println(dado.toLowerCase());
        System.out.println(dado.length());
        System.out.println(dado.substring(0,5));
        System.out.println(dado.replaceAll("World", "Mundo"));
    }
}
```

```
run:|
Hello World
HELLO WORLD
hello world
ll
Hello
Hello Mundo
BUILD SUCCESSFUL (total time: 0 seconds)
```

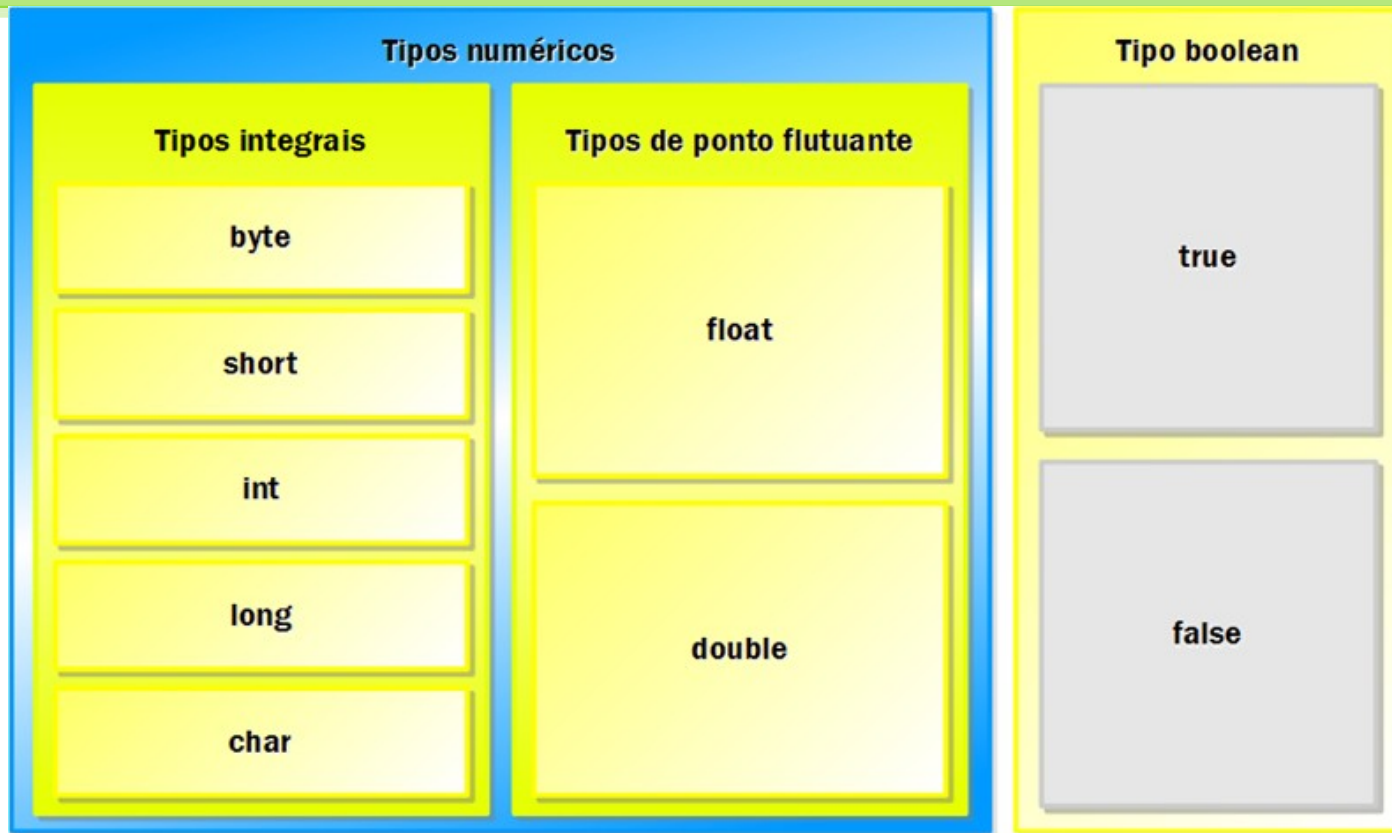
# Garbage Collector (Coletor de lixo)

- \* É um gerenciamento automático da memória;
- \* Em outros compiladores como o C, esse processo é feito manualmente pelo desenvolvedor. Já no Java tal processo é automático, deixando a memória livre caso a mesma não seja mais usada;
- \* Os objetos são criados quando necessário, e quando não estão mais em uso, a JVM através do GC remove automaticamente os objetos e libera memória;
- \* Ideal para jogos e outros apps para não ocupar memória sem uso

# Palavras chave

abstract	continue	for	new	switch
assert	default	goto	package	synchronized
boolean	do	if	private	this
break	double	implements	protected	throw
byte	else	import	public	throws
case	enum	instanceof	return	transient
catch	extends	int	short	try
char	final	interface	static	void
class	finally	long	strictfp	volatile
const	float	native	super	while

# Tipos de dados primitivos



# Tipos de dados primitivos – Tipos numéricos

Tipo	Faixa	Bytes
byte	-128 a 127	1
short	-32768 a 32767	2
int	-2147483648 a 2147483647	4
long	-9223372036854775808 a 9223372036854775807	8
char	0 a 65535	2

# Tipos de dados primitivos – Tipos numéricos

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        byte x;
        x=33;
        System.out.println(x);
    }
}
```

# Tipos de dados primitivos – Tipos numéricos

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        byte x=33;
        System.out.println(x);
    }
}
```



# Tipos de dados primitivos – Tipos numéricos

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        byte x=330;
        System.out.println(x);
    }
}
```

?

# Tipos de dados primitivos – Tipos numéricos

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        short x=3300;
        System.out.println(x);
    }
}
```

# Tipos de dados primitivos – Tipos numéricos

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        int x=100000;
        System.out.println(x);
    }
}
```

# Tipos de dados primitivos – Tipos numéricos

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        long x=9999999999;
        System.out.println(x);
    }
}
```

# Tipos de dados primitivos – Tipos numéricos

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        char x='a';
        System.out.println(x);
    }
}
```

# Tabela ASCII

*	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	TAB	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2		!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	

# Tipos de dados primitivos – Unicode

```
package javaapplication1;  
  
public class JavaApplication1  
{  
    public static void main(String[] args)  
    {  
        char x='\u0058';  
        System.out.println(x);  
    }  
}
```

# Tipos de dados primitivos – Typecast

```
package javaapplication1;
```

```
public class JavaApplication1
```

```
{
```

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

```
    {
```

```
        int number1 = 66;
```

```
        char chars1 = (char)number1;
```

```
        int number2 = 76;
```

```
        char chars2 = (char)number2;
```

```
    }
```

```
    int number3 = 79;
```

```
    char chars3 = (char)number3;
```

```
int number4 = 71;
```

```
char chars4 = (char)number4;
```

```
System.out.println(chars1);
```

```
System.out.println(chars2);
```

```
System.out.println(chars3);
```

```
System.out.println(chars4);
```

```
}
```



# Tipos de dados primitivos

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        float num1=3.14592f;
        System.out.println(num1);
    }
}
```

# Tipos de dados primitivos

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        double num1=3.14592;
        System.out.println(num1);
    }
}
```

# Tipos de dados primitivos

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        boolean y=true;
        System.out.println(y);
    }
}
```

# Operadores de atribuição +=, -=, \*=, /=, %=

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        int x=10;

        x+=4;
        System.out.println("Soma= " + x);

        x-=6;
        System.out.println("Subtração= " + x);
    }
}
```

```
x*=6;
System.out.println("Produto= " + x);

x/=3;
System.out.println("Divisão= " + x);

x%=5;
System.out.println("Resto da divisão= " + x);
```



# Operadores de igualdade e diferença ==, !=

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        int x=10,y=20;
        boolean z;

        z = x==y;
        System.out.println("x==y? " + z);

        z = x!=y;
        System.out.println("x!=y? " + z);
    }
}
```

# Operadores relacionais >, >=, <, <=

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        int x=10,y=20;
        boolean z;

        z = x>y;
        System.out.println("x>y? " + z);

        z = x>=y;
        System.out.println("x>=y? " + z);
    }
}
```

```
z = x<y;
System.out.println("x<y? " + z);

z = x<=y;
System.out.println("x<=y? " + z);
}
```

# Operador de incremento e decremento ++ --

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        int x=10;
        System.out.println(x++);
        System.out.println(++x);
        System.out.println(--x);
        System.out.println(x--);
        System.out.println(x);
    }
}
```

# Trabalhando com base binária

```
package javaapplication1;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        byte x=0b00001011;
        System.out.println(x);
        System.out.println(Integer.toString(x,2));
        System.out.println(Integer.toString(x,8));
        System.out.println(Integer.toString(x,16));
    }
}
```



# Entrada de dados

```
package javaapplication1;

import java.util.Scanner;

public class JavaApplication1
{
    public static void main(String[] args)
    {
        String nome;
        Scanner inp = new Scanner(System.in);
        System.out.println("Qual o seu nome? ");
        nome = inp.next();
        System.out.println("Seja bem-vindo " + nome + " ao Java!");
    }
}
```