



CURSO TECNICO EM INFORMÁTICA
DISCIPLINA: LINGUAGEM DE PROGRAMAÇÃO

INSTALAÇÃO DO JAVA JKD E DO JCREATOR

Vídeo 1 da aula: <https://www.youtube.com/watch?v=6Q-jaWUBIrg>
Continuação:

Prof. André Aparecido da Silva.

Disponível em: <http://www.oxnar.com.br/aulas/java>

Acesse o site da Oracle

<https://www.oracle.com/technetwork/pt/java/javase/downloads/index.html>

The screenshot shows a web browser window with the address bar displaying the URL <https://www.oracle.com/technetwork/pt/java/javase/downloads/index.html>. The browser's address bar also shows "Não seguro" (Not secure) and the Oracle logo. The page content includes a navigation menu on the left with links to Java SE, Soporte para Java SE, Java Embedded, Java EE, Java ME, Java FX, Java DB, Web Tier, and Comunidade. The main content area is titled "Java SE Downloads" and features two download buttons: "DOWNLOAD" for Java Platform (JDK) 8u111 / 8u112 and "DOWNLOAD" for NetBeans com JDK 8. Below these, there is a section for "Java Platform, Standard Edition" with a heading "Java SE 8u111 / 8u112" and a paragraph stating that Java SE 8u111 includes important security fixes and that Oracle strongly recommends that all Java SE 8 users upgrade to this release. A link "Learn more" is provided. A yellow box highlights an "Important planned change for MD5-signed JARs" starting with the April Critical Patch Update releases, planned for April 18 2017, where all JRE versions will treat JARs signed with MD5 as unsigned. A link "Learn more and view testing instructions" is provided. At the bottom, there is a link to "JDK" and a footer with the URL www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-213315... and a link to "allation Instructions".

Java SE Downloads

Resumo Downloads Documentação Comunidade Tecnologia Formação

Java SE
Soporte para Java SE
Java Embedded
Java EE
Java ME
Java FX
Java DB
Web Tier
Comunidade

Java Platform (JDK) 8u111 / 8u112

NetBeans com JDK 8

Java Platform, Standard Edition

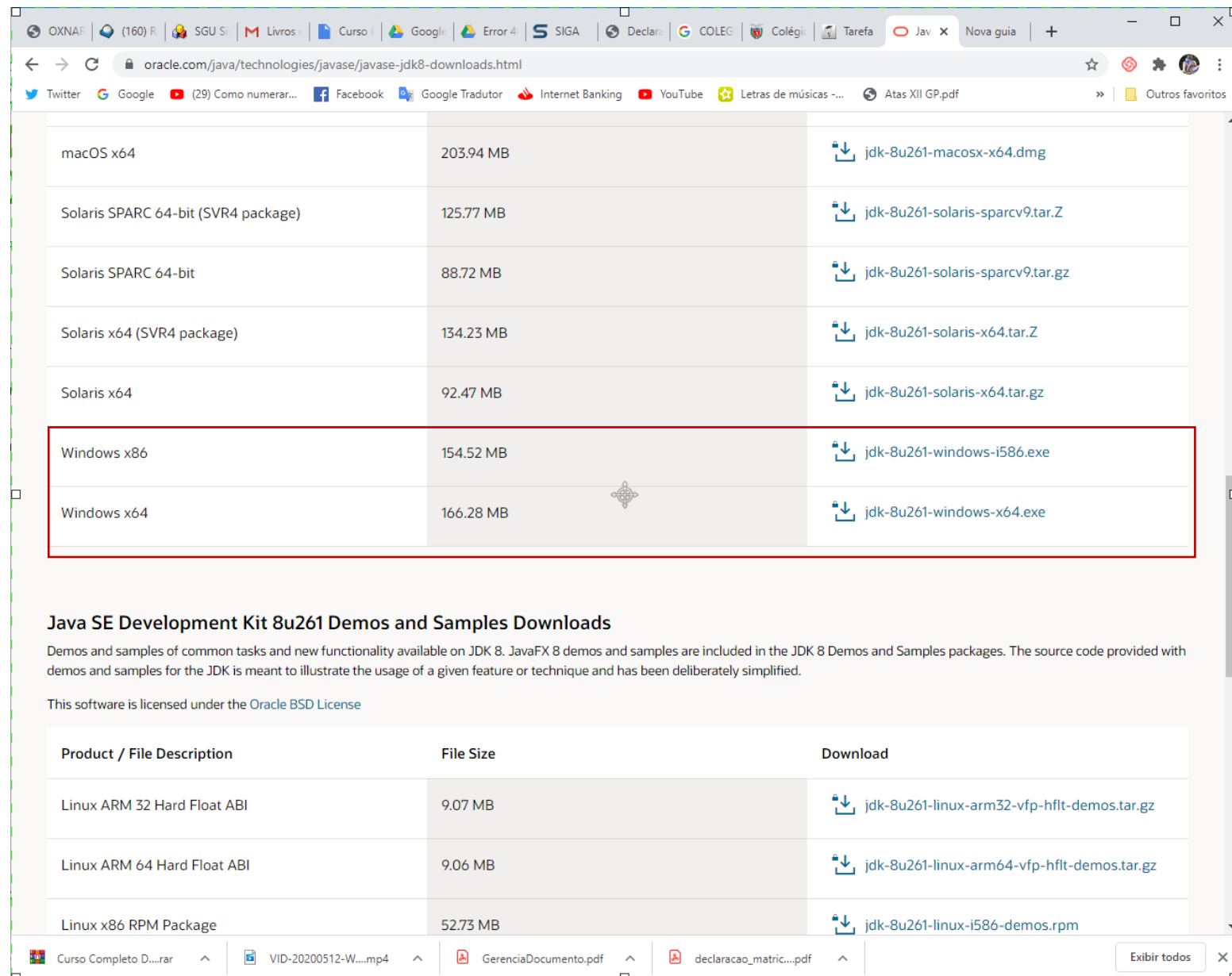
Java SE 8u111 / 8u112
Java SE 8u111 includes important security fixes. Oracle strongly recommends that all Java SE 8 users upgrade to this release. Java SE 8u112 is a patch-set update, including all of 8u111 plus additional features (described in the release notes).
[Learn more](#)

Important planned change for MD5-signed JARs
Starting with the April Critical Patch Update releases, planned for April 18 2017, all JRE versions will treat JARs signed with MD5 as unsigned. [Learn more and view testing instructions](#).
For more information on cryptographic algorithm support, please check the [JRE and JDK Crypto Roadmap](#).

JDK

www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-213315... [allation Instructions](#)

Escolha a versão voltada para o seu sistema operacional



oracle.com/java/technologies/javase/javase-jdk8-downloads.html

macOS x64	203.94 MB	jdk-8u261-macosx-x64.dmg
Solaris SPARC 64-bit (SVR4 package)	125.77 MB	jdk-8u261-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	88.72 MB	jdk-8u261-solaris-sparcv9.tar.gz
Solaris x64 (SVR4 package)	134.23 MB	jdk-8u261-solaris-x64.tar.Z
Solaris x64	92.47 MB	jdk-8u261-solaris-x64.tar.gz
Windows x86	154.52 MB	jdk-8u261-windows-i586.exe
Windows x64	166.28 MB	jdk-8u261-windows-x64.exe

Java SE Development Kit 8u261 Demos and Samples Downloads

Demos and samples of common tasks and new functionality available on JDK 8. JavaFX 8 demos and samples are included in the JDK 8 Demos and Samples packages. The source code provided with demos and samples for the JDK is meant to illustrate the usage of a given feature or technique and has been deliberately simplified.

This software is licensed under the [Oracle BSD License](#)

Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	9.07 MB	jdk-8u261-linux-arm32-vfp-hflt-demos.tar.gz
Linux ARM 64 Hard Float ABI	9.06 MB	jdk-8u261-linux-arm64-vfp-hflt-demos.tar.gz
Linux x86 RPM Package	52.73 MB	jdk-8u261-linux-i586-demos.rpm

Após a instalação será necessário setar as variáveis de ambiente

Windows

Windows 10 e Windows 8

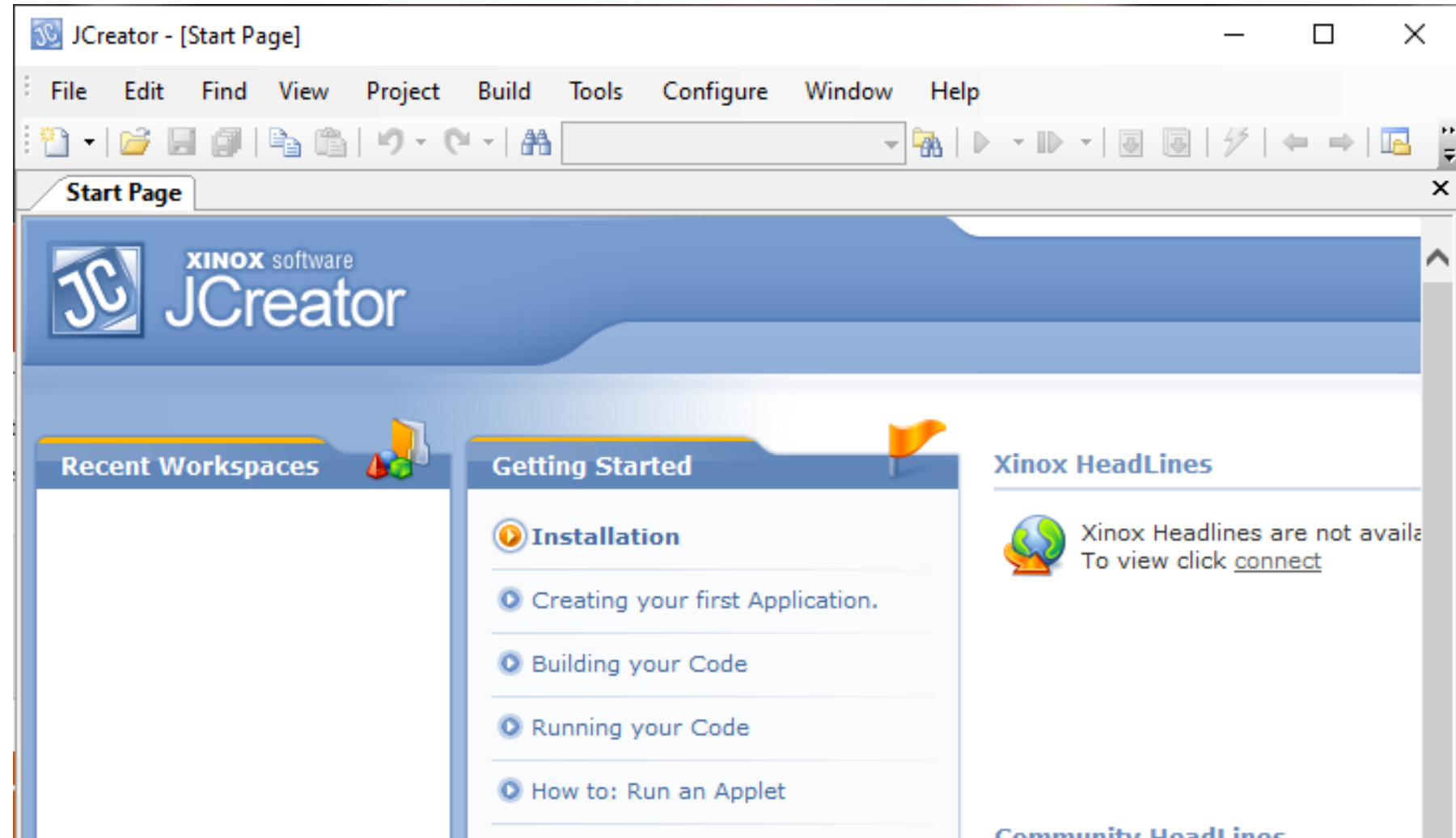
1. Em Pesquisar, procure e selecione: Sistema (Painel de Controle)
2. Clique no link Configurações avançadas do sistema.
3. Clique em Variáveis de Ambiente. Na seção Variáveis do Sistema, localize a variável de ambiente PATH e selecione-a. Clique em Editar. Se a variável de ambiente PATH não existir, clique em Novo.
4. Na janela Editar Variável de Sistema (ou Nova Variável de Sistema), especifique o valor da variável de ambiente PATH. Clique em OK. Feche todas as janelas restantes clicando em OK.
5. Reabra a janela Prompt de comando e execute o código Java.

Após a instalação será necessário setar as variáveis de ambiente

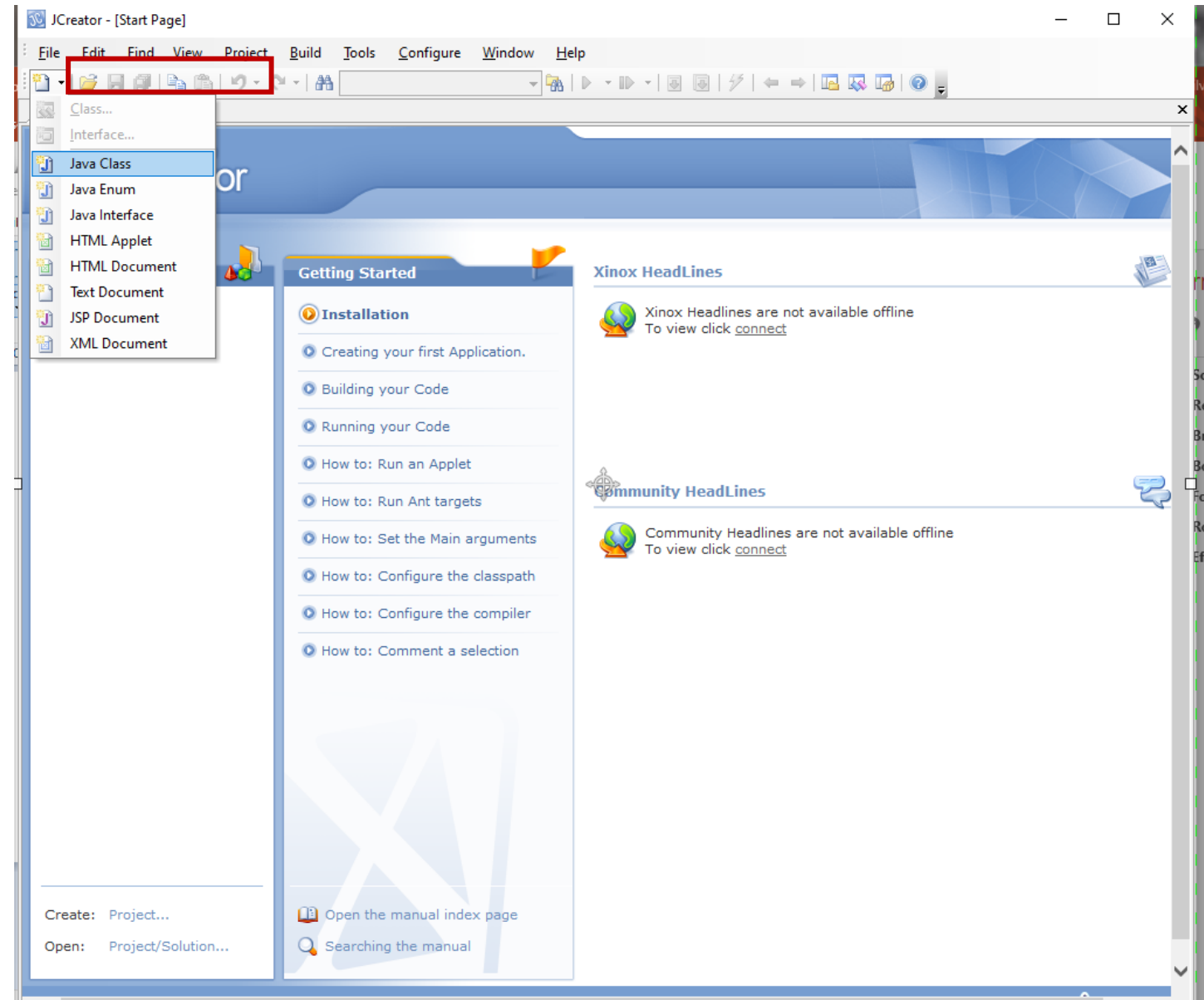
Aperte as teclas Windows + (pause break para acessar as configurações avançadas do sistema.

Baixar e instalar o JCreator

<http://www.jcreator.org/download.htm>



Agora vamos criar a nosso primeiro programa no java



Nosso código

```
public class Ola
{
    public static void main (String args [])
    {
        System.out.println("OLÁ PESSOAL DO CEP");
    }
}
```

Ola2.java

```
public class Ola2
{
    public Ola2()
    {
        System.out.println("OLÁ PESSOAL DO CEP");
    }
    public static void main(String args[])
    {
        new Ola2();
    }
}
```

Algumas considerações

// Podem ser utilizados para comentários de apenas uma linha

/*

Comentários mais use barra asterisco para começar e asterisco barra da finalizar;

*/

Algumas considerações

Início e fim de blocos de instruções são demarcados por abre e fecha chaves

{

}

Algumas considerações

As instruções são finalizadas com ponto e virgula

```
public class OxnarX
{
    public static void main (String[] args)
    {
        System.out.println("Falta ponto e vírgula")
    }
}
```

Erro: X.java:4: ';' expected } ^ 1 error

PALAVRAS RESERVADAS

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 PRIMITIVOS

<i>Tipo</i>	<i>Tamanho (bits)</i>	<i>Valor Mínimo</i>	<i>Valor Máximo</i>	<i>Sem Sinal</i>
boolean	1	false	true	X
char	16	0	$2^{16} - 1$	X
byte	8	-2^7	$2^7 - 1$	
short	16	-2^{15}	$2^{15} - 1$	
int	32	-2^{31}	$2^{31} - 1$	
long	64	-2^{63}	$2^{63} - 1$	
float	32			
double	64			

TIPOS PRIMITIVOS - Lógico

<i>Tipo</i>	<i>Tamanho (bits)</i>	<i>Valor Mínimo</i>	<i>Valor Máximo</i>	<i>Sem Sinal</i>
boolean	1	false	true	X

```
boolean a = false;  
boolean b = true;
```

Assume o valor 0 para false e 1 para verdadeiro;

TIPOS PRIMITIVOS - char

<i>Tipo</i>	<i>Tamanho (bits)</i>	<i>Valor Mínimo</i>	<i>Valor Máximo</i>	<i>Sem Sinal</i>
char	16	0	$2^{16} - 1$	X

char tipo = 'a';

char categoria = '2';

Armazena apenas 1 caractere

TIPOS PRIMITIVOS - byte

<i>Tipo</i>	<i>Tamanho (bits)</i>	<i>Valor Mínimo</i>	<i>Valor Máximo</i>	<i>Sem Sinal</i>
byte	8	-2^7	$2^7 - 1$	

byte Idade = 13;

byteCodigo = 7;

Armazena números de -128 a 127

TIPOS PRIMITIVOS - short

<i>Tipo</i>	<i>Tamanho (bits)</i>	<i>Valor Mínimo</i>	<i>Valor Máximo</i>	<i>Sem Sinal</i>
byte	8	-2^7	$2^7 - 1$	

		Valores possíveis				
<i>Tipos</i>	<i>Primitivo</i>	<i>Menor</i>	<i>Maior</i>	<i>Valor Padrão</i>	<i>Tamanho</i>	<i>Exemplo</i>
Inteiro						
	short	-32768	32767	0	16 bits	short ex2 = (short)1;

Armazena números de -32768 a 32767

TIPOS PRIMITIVOS - short

```
short Numero_Aluno = 736;  
short Numero_Romano = 1257;
```

TIPOS PRIMITIVOS - int

		Valores possíveis				
Tipos	Primitivo	Menor	Maior	Valor Padrão	Tamanho	Exemplo
Inteiro						
	int	-2.147.483.648	2.147.483.647	0	32 bits	int ex3 = 1;

```
int Votos_Validos =132888;  
Int Votos_Nulos = 328222;
```

Armazena números de -2.147.483.648 a 2.147.483.647

TIPOS PRIMITIVOS - long

<i>Tipo</i>	<i>Tamanho (bits)</i>	<i>Valor Mínimo</i>	<i>Valor Máximo</i>	<i>Sem Sinal</i>
long	64	-2^{63}	$2^{63} - 1$	

-9.223.372.036.854.775.808L e 9.223.372.036.854.775.807L

TIPOS PRIMITIVOS - long

<i>Tipo</i>	<i>Tamanho (bits)</i>	<i>Valor Mínimo</i>	<i>Valor Máximo</i>	<i>Sem Sinal</i>
long	64	-2^{63}	$2^{63} - 1$	

-9.223.372.036.854.775.808L e 9.223.372.036.854.775.807L

TIPOS PRIMITIVOS – float /double

<i>Tipo</i>	<i>Tamanho (bits)</i>	<i>Valor Mínimo</i>	<i>Valor Máximo</i>	<i>Sem Sinal</i>
float	32			
double	64			

Exemplos:

```
float Salario = 44521.27;  
double divida = 63123.81;
```

E as String

Diferentemente dos tipos primitivos `int`, `double`, `char` e `boolean` o tipo `String` é "objetos". O tipo `String`, com `S` maiúsculo, é um dos objetos mais utilizados.

Exemplos:

```
String Nome = "André Aparecido da Silva";
```

```
String Turma = "Turma 2º TI";
```

```
String Site = "www.oxnar.com.br";
```

```
String Numero_Romano = "MMMCMXCIX";
```

Variável final

- Uma variável final uma vez atribuído um valor a ela esse valor não pode ser modificado e se não me engano você pode atribuir um valor a ela apenas na sua declaração ou no construtor da classe.

```
final String Nome = “André Aparecido da Silva”;
```

```
final double pi = 3.141592653589;
```

```
public class Exemplo_Tipos2
{

    boolean    Status = false;
    char        Turma = 'a';
    byte        Idade = 40;
    double      Salario = 7000.45;
    String      Nome = "André Aparecido da Silva";

    public Exemplo_Tipos2 ()
    {
        System.out.println("IMPRIMINDO VARIÁVEIS");
        System.out.println("MEU NOME É: "+Nome );
        System.out.println("MEU STATUS É: "+Status );
        System.out.println("MINHA IDADE É: "+ Idade);
        System.out.println("MINHA SALÁRIO É: "+ Salario/4);
        System.out.println("");
        System.out.println("");
        System.out.println("");
        System.out.println("");

        //Metodo_Mostra_Tipos_Java();

    }

    public static void main (String args [])
    {
        new Exemplo_Tipos2 ();
    }
}
```