

TALLER DE LENGUAJE DE PROGRAMACIÓN I

Entorno de CONSOLA y GRÁFICOS

UNIDAD 1

Objeto String – java.lang

Objeto Date – java.util

Objeto Calendar – java.util

Paquete java.time

JAVA

OBJETIVO

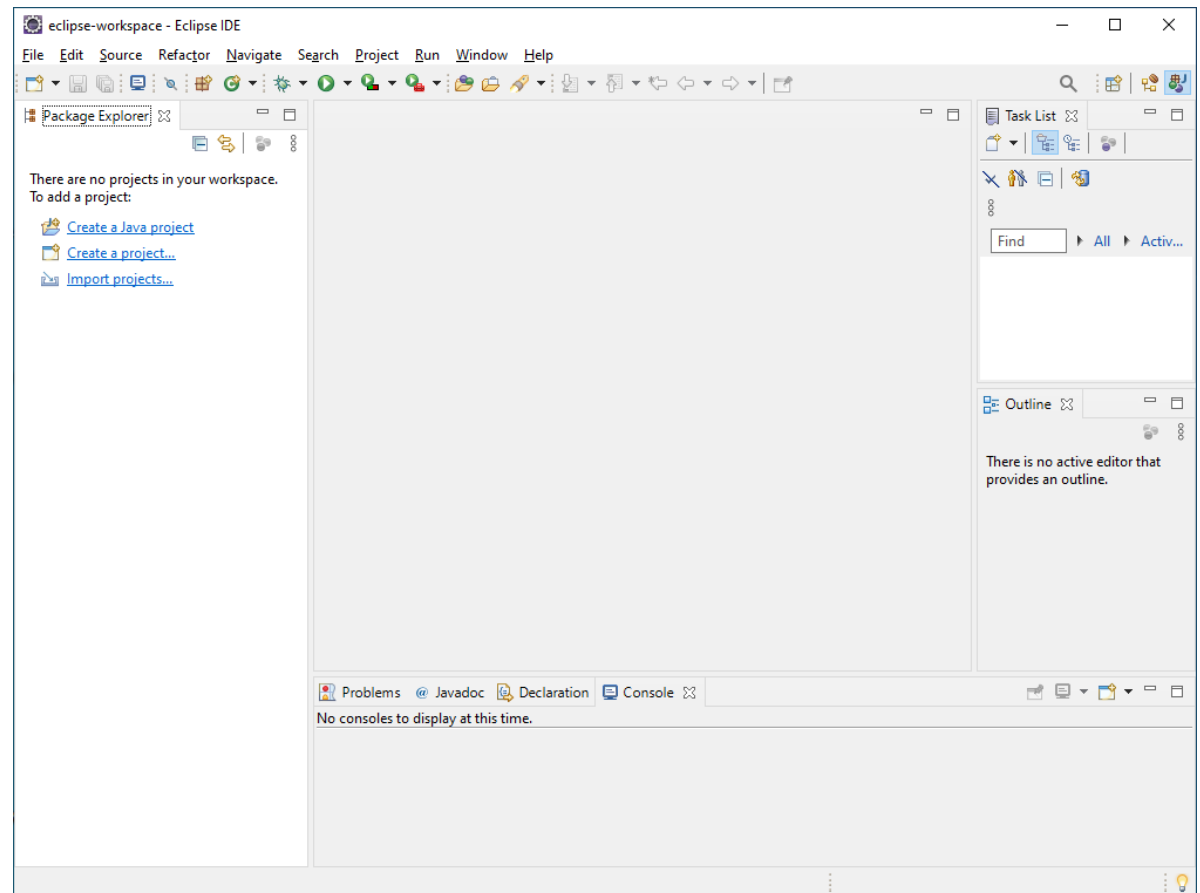
- Ejecución del IDE Eclipse
- Clase (Objeto) String de JAVA
- Paquetes API JAVA
 - Paquete java.lang
- Objeto Date – java.util, Calendar – java.util
- Paquete java.time

Trabajo Independiente

- Crear clase con el IDE Eclipse.
- Consultar la clase DATE
- Crear una aplicación, donde se utilicen todos los tipos de datos primitivos más el objeto String y DATE
- Consultar la definición de constantes, operadores lógicos.

IDE Eclipse

Cargar Eclipse.



IDE Eclipse, continuación

- Crear una aplicación que opere dos números enteros, mostrando el resultado en la consola del IDE Eclipse
1. Crear un proyecto en Eclipse.
 2. Digitar las instrucciones de la aplicación.
 - Refundición.
 3. Ver como queda el encarpetao en la ubicación del proyecto.

IDE Eclipse, continuación

Refundición.

```
public class Operaciones
{
    public static void main(String[] args)
    {
        int num1 = 0;
        int num2 = 0;
        int resultado = 0;

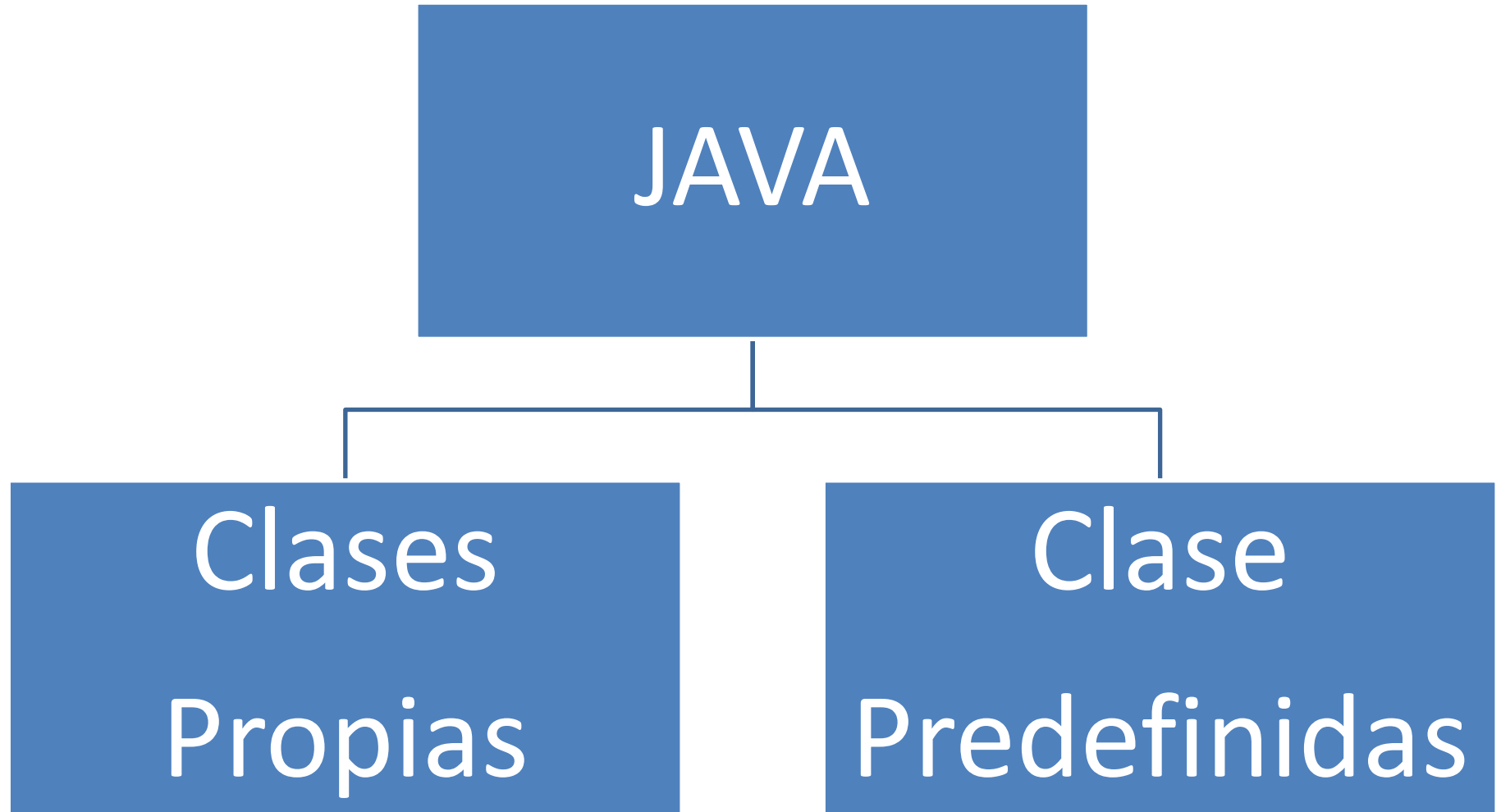
        num1 = 10;
        num2 = 3;
        resultado = num1 * num2;
        System.out.println("Resultado: " + resultado);
        resultado = (int) Math.sqrt(10);
        System.out.println("Resultado: " + resultado);

        double resultado2 = Math.sqrt(10);
        System.out.println("Resultado2: " + resultado2);
    }
}
```

Redefinir el tipo de dato

Resultado valor double

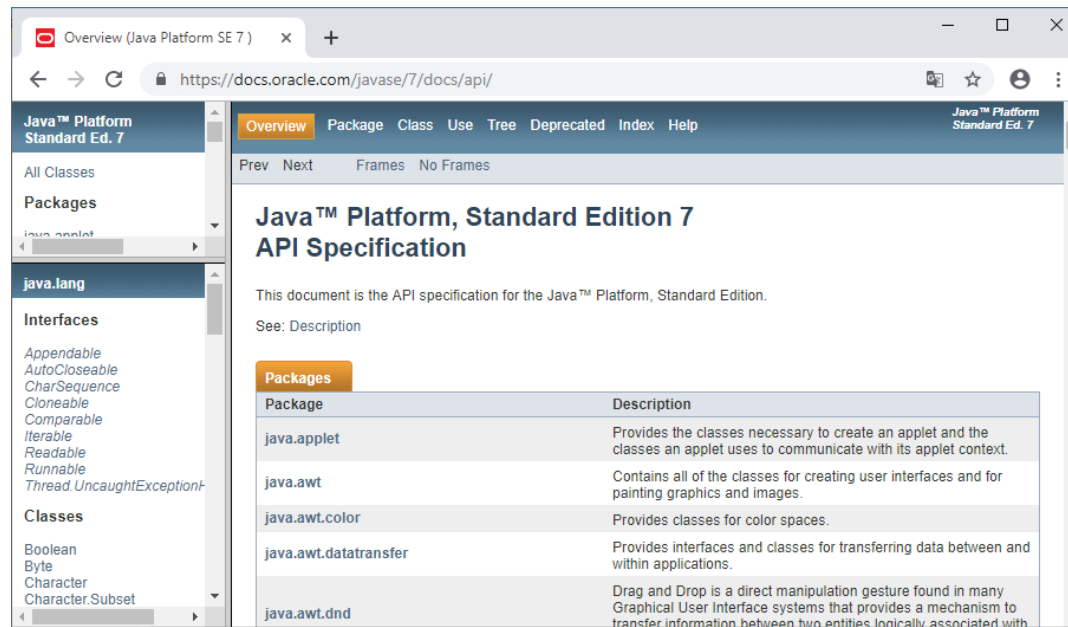
Objeto String de JAVA



Objeto String de JAVA, continuación

API de JAVA

- **API** es una sigla en ingles **A**pplication **P**rogramming **I**nterface - Interfaz de Programación de Aplicaciones. El concepto hace referencia a los procesos, las funciones y los métodos que brinda una **determinada biblioteca de programación** a modo de capa de abstracción para que sea empleada por otro programa informático.



The screenshot shows the Oracle Java Platform SE 7 API Specification page. The browser address bar displays <https://docs.oracle.com/javase/7/docs/api/>. The page title is "Java™ Platform, Standard Edition 7 API Specification". The left sidebar contains a navigation menu with "All Classes", "Packages", and "java.lang" selected. Under "Interfaces", a list of interfaces is shown, including `Appendable`, `AutoCloseable`, `CharSequence`, `Cloneable`, `Comparable`, `Iterable`, `Readable`, `Runnable`, and `Thread.UncaughtExceptionHandler`. Under "Classes", a list of classes is shown, including `Boolean`, `Byte`, `Character`, and `Character.Subset`. The main content area shows the "Overview" tab selected, with a table of packages and their descriptions.

Package	Description
<code>java.applet</code>	Provides the classes necessary to create an applet and the classes an applet uses to communicate with its applet context.
<code>java.awt</code>	Contains all of the classes for creating user interfaces and for painting graphics and images.
<code>java.awt.color</code>	Provides classes for color spaces.
<code>java.awt.datatransfer</code>	Provides interfaces and classes for transferring data between and within applications.
<code>java.awt.dnd</code>	Drag and Drop is a direct manipulation gesture found in many Graphical User Interface systems that provides a mechanism to transfer information between two entities logically associated with

Objeto String de JAVA, continuación

Clase String de JAVA

Paquete por defecto.

The screenshot shows the Java Platform SE 7 API documentation for the `String` class. The browser address bar shows the URL `https://docs.oracle.com/javase/7/docs/api/`. The left sidebar lists various packages and classes, with `String` highlighted under the `java.lang` package. The main content area displays the `String` class details, including its inheritance hierarchy (`java.lang.Object` and `java.lang.String`), implemented interfaces (`Serializable`, `CharSequence`, `Comparable<String>`), and the class declaration: `public final class String extends Object implements Serializable, Comparable<String>, CharSequence`. The text explains that `String` represents character strings and that all string literals in Java programs are implemented as instances of this class. It also notes that `String` objects are immutable and can be shared. An example code snippet shows `String str = "abc";` and its equivalent array representation: `char data[] = {'a', 'b', 'c'};`.

Objeto String de JAVA, continuación

Clase String de JAVA

- El tipo de dato String no es un tipo de dato primitivo, es un clase (Objeto).
- Como definir la variable tipo String



Se instancia el objeto de tipo String

String cadena = "Alguna cadena";

Objeto String de JAVA, continuación

Clase String de JAVA y sus métodos

- `length()` → Longitud de una cadena.
- `charAt()` → Posición de una carácter en una cadena
- `substring()` → Extraer cadenas de una cadena
- `equals()` → Compara dos cadenas
- `equalsIgnoreCase()` → Compara dos cadenas sin tener en cuenta mayúsculas ni minúsculas.
- `toUpperCase()` → Convierte la cadena a mayusculas

Objeto String de JAVA, continuación

Clase String de JAVA y sus métodos **length()**

```
public class longitud
{
    public static void main(String[] args)
    {
        //Utilización de la clase String del paquete java.lang

        String cadena = "Jose Botero";

        System.out.println("Longitud de caracteres: " + cadena.length());

        int longitud = 0;

        longitud = cadena.length();
        System.out.println("Longitud de caracteres: " + longitud);
    }
}
```

Objeto String de JAVA, continuación

Clase String de JAVA y sus métodos **charAt()**

```
public class charAt
```

```
{
```

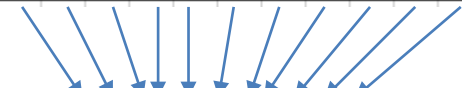
```
    public static void main(String[] args)
```

```
    {
```

```
        //Utilización de la clase String del paquete java.lang
```

0	1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	---	----

String cadena = "Jose Botero";



```
        System.out.println("Caracter en la posición 1 de la cadena: " +  
        cadena.charAt(1) );
```

```
    }
```

```
}
```

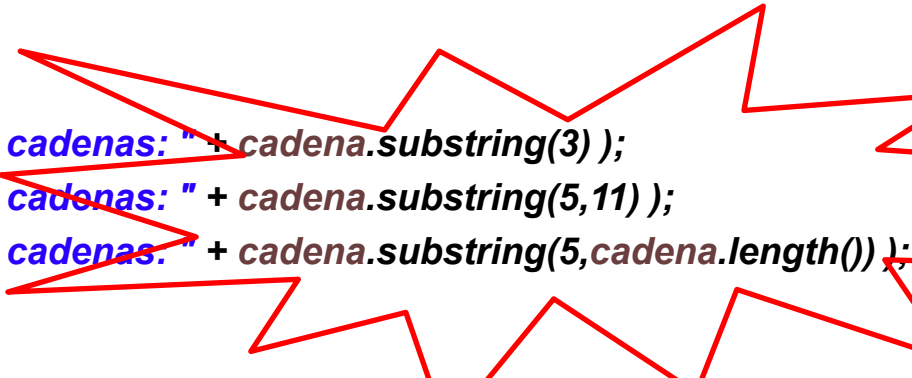
Objeto String de JAVA, continuación

Clase String de JAVA y sus métodos **substring()**

```
public class longitud
{
    public static void main(String[] args)
    {
        //Utilización de la clase String del paquete java.lang

        String cadena = "Jose Botero";

        System.out.println("Subextraer cadenas: " + cadena.substring(3) );
        System.out.println("Subextraer cadenas: " + cadena.substring(5,11) );
        System.out.println("Subextraer cadenas: " + cadena.substring(5,cadena.length()) );
    }
}
```



Objeto String de JAVA, continuación


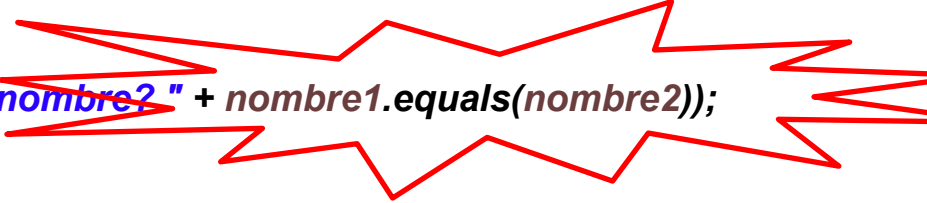
Clase String de JAVA y sus métodos `equals()`

```
public class longitud
{
    public static void main(String[] args)
    {
        //Utilización de la clase String del paquete java.lang

        String nombre1, nombre2;
        nombre1 = "Cristina";
        nombre2 = "Cristina";

        System.out.println("Son iguales los nombre? " + nombre1.equals(nombre2));

        System.out.println("Son iguales los nombre? " + nombre1.equalsIgnoreCase(nombre2));
    }
}
```



Objeto String de JAVA, continuación

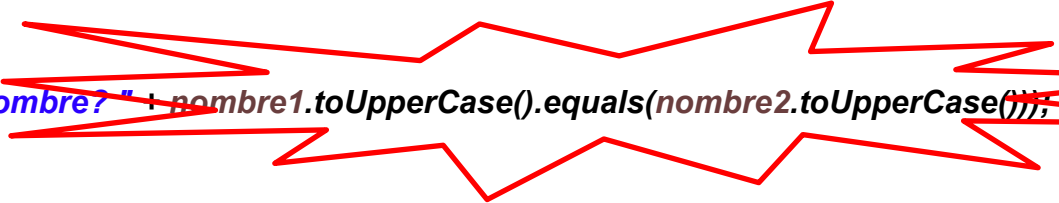
Clase String de JAVA y sus métodos toUpperCase()

```
public class longitud
{
    public static void main(String[] args)
    {
        //Utilización de la clase String del paquete java.lang

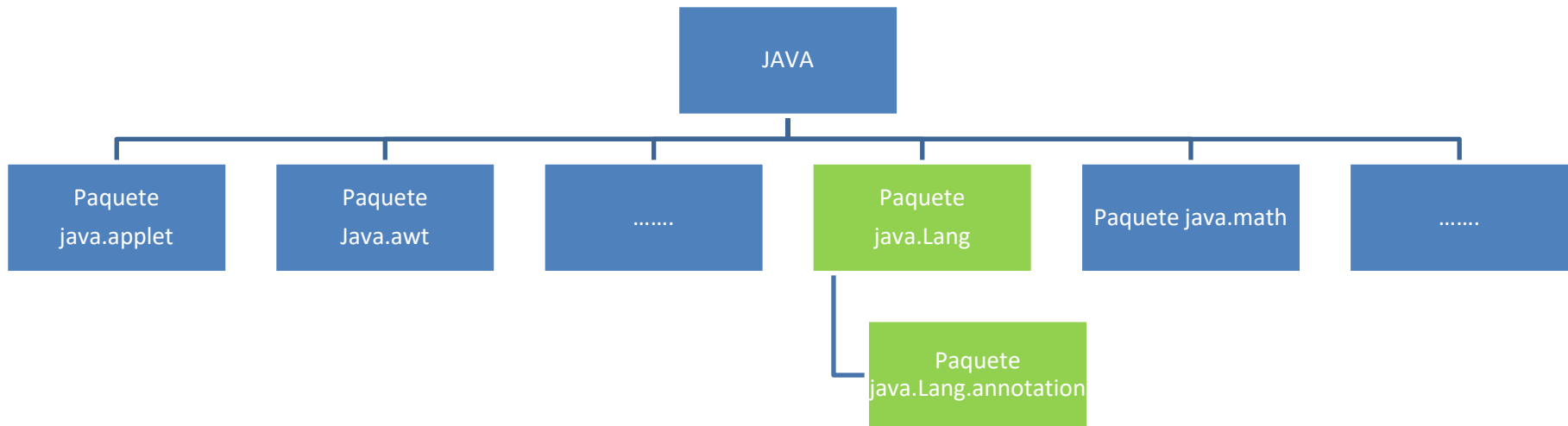
        String nombre1, nombre2;
        nombre1 = "Cristina";
        nombre2 = "Cristina";

        System.out.println("Son iguales los nombre? " + nombre1.toUpperCase().equals(nombre2.toUpperCase()));

    }
}
```



Paquetes en JAVA



Paquetes en JAVA, continuación

Directiva import

Es utilizada para trabajar con clases que no están incluidas en el paquete por defecto de JAVA, el paquete java.lang

En el siguiente ejemplo capturaremos dos valores enteros y mostraremos el resultado de multiplicarlos.

1. Utilizar la clase Scanner y los métodos nextLine(), nextInt() y nextDouble()
2. Utilizar la clase JOptionPane y el método showInputDialog()

Paquetes en JAVA, continuación

Directiva **import** paquete java.util clase Scanner

```
import java.util.Scanner;
```

```
public class directivas
```

```
{
```

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

```
    {
```

```
        //Utilizamos la clase por defecto java.lang
```

```
        String cadena;
```

```
        cadena = "Resultado de la multiplicación es: ";
```

```
        //Utilizamos una directiva para java.util
```

```
        Scanner wdatos = new Scanner(System.in);
```

```
        System.out.println("Digite su nombre:");
```

```
        String wnombre = wdatos.nextLine();
```

```
        System.out.println("Digite su edad:");
```

```
        int wedad = wdatos.nextInt();
```

```
        System.out.println("Hola " + wnombre + " ya casi tienes " + wedad);
```

```
    }
```

```
}
```

Paquetes en JAVA, continuación

Directiva `import` paquete `javax.swing.JOptionPane`

```
import javax.swing.*;
```

```
public class directivas
```

```
{
```

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

```
    {
```

```
        String nombre = JOptionPane.showInputDialog("Digite su nombre");
```

```
        String edad = JOptionPane.showInputDialog("Digite su edad");
```

```
        JOptionPane.showMessageDialog(null, "Hola " + nombre + " solo tienes " + edad);
```

```
    }
```

```
}
```

Trabajo Independiente

- Crear clases con el IDE Eclipse.
- Crear una aplicación, donde se utilicen todos los tipos de datos primitivos más el objeto String
- Investigar e implementar la clase Date, Calendar, Paquete java.time
- Consultar la definición de variables constantes y operadores lógicos.