


Java



JAVA és un llenguatge de programació orientat a objectes, multiplataforma.

Es pot considerar un llenguatge compilat i interpretat.

Origen: Empresa Sun Microsystem (James Gosling i Patrick Naughton), comprat posteriorment per Oracle (gestor de base de dades).

Característiques:

- **Senzill** (característiques més enutjoses d'altres llenguatges, i en general ocupa poc espai)
- **Orientat a objectes** (un dels que més)
- **Distribuït** (s'adapta bé a l'hora de programar en xarxa)
- **Segur** (un dels més segurs, no 100%)
- Neutre respecte arquitectura (**multiplataforma**)
- **Compilat i interpretat**
- **Multifil** (es poden programar diversos processos en el mateix programa)

Llenguatge compilat i no interpretat

```
/* Programa: Suma y multiplicación de dos números (Solución) */

#include <conio.h>
#include <stdio.h>

int main()
{
    int n1, n2, producto, suma;

    printf( "\n  Introduzca primer numero (entero): " );
    scanf( "%d", &n1 );
    printf( "\n  Introduzca segundo numero (entero): " );
    scanf( "%d", &n2 );

    suma = n1 + n2;
    producto = n1 * n2;

    printf( "\n  La suma es: %d", suma );
    printf( "\n\n  La multiplicación es: %d", producto );

    getch(); /* Pausa */

    return 0;
}
```

CODI FONT

COMPILAR
(pas de codi font a
codi màquina, per
exemple
.exe en windows)



0 0 0 0	1 1 1 0
0 0 0 0	0 0 0 0
0 0 1 1	1 1 1 0
0 1 1 0	0 1 0 0
0 0 0 0	1 1 0 0
1 0 0 1	0 0 0 1
1 1 0 0	0 0 1 0
0 0 0 0	0 0 1 0
0 0 1 0	0 0 0 0

CODI MÀQUINA

PER DIFERENTS PLATAFORMES:

- Windows
- Linux
- Mac

S'han de crear versions diferents per a poder executar-ho en cadascuna de les plataformes

JRE:

Java Runtime Environment (Entorn d'execució de Java).

També conegut com **JVM** (Màquina virtual de Java, java virtual machine)

Principal característica de Java : multiplataforma

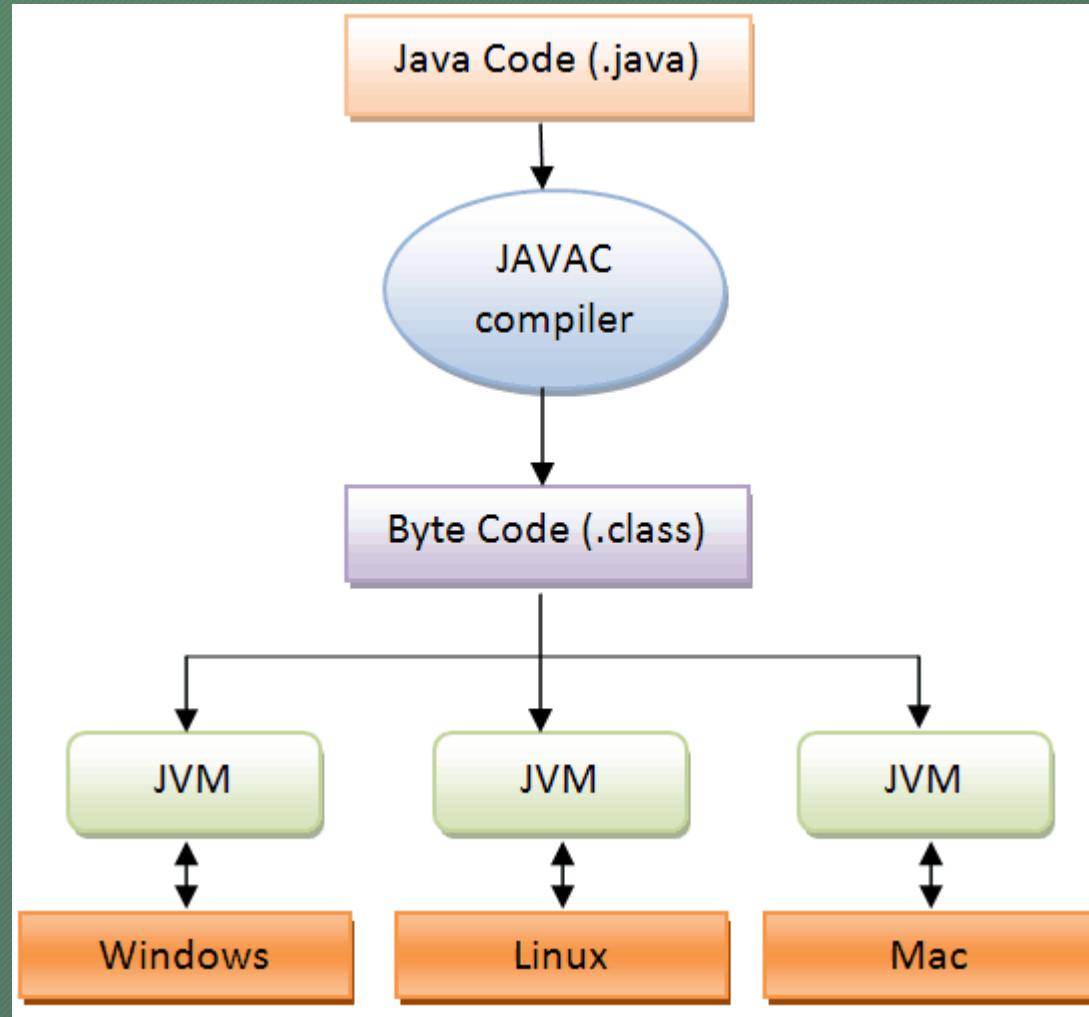
Que permet que inicialment un arxiu java sigui compilat i després interpretat per la JVM.

JRE

Java es **multiplataforma**:

El compilador de java és javac.

Se li afegeix un nivell entre el programa executable i el codi màquina.



JVM de Java per a casa plataforma de hardware específic

En **compilar** obtenim un arxiu intermedi de bytecodes , codi intermedi entre el codi font i el codi màquina.

El codi per a la JVM és de tipus .class

JVM **interpreta** l'arxiu de bytecodes (.class)

**Write once,
run every
where**

IDE

Entorn de desenvolupament integrat.

- Aplicació de software que facilita als programadors a desenvolupar codi de software de manera eficient.
- Combina eines de desenvolupador comuns en una única interfície gràfica d'usuari.

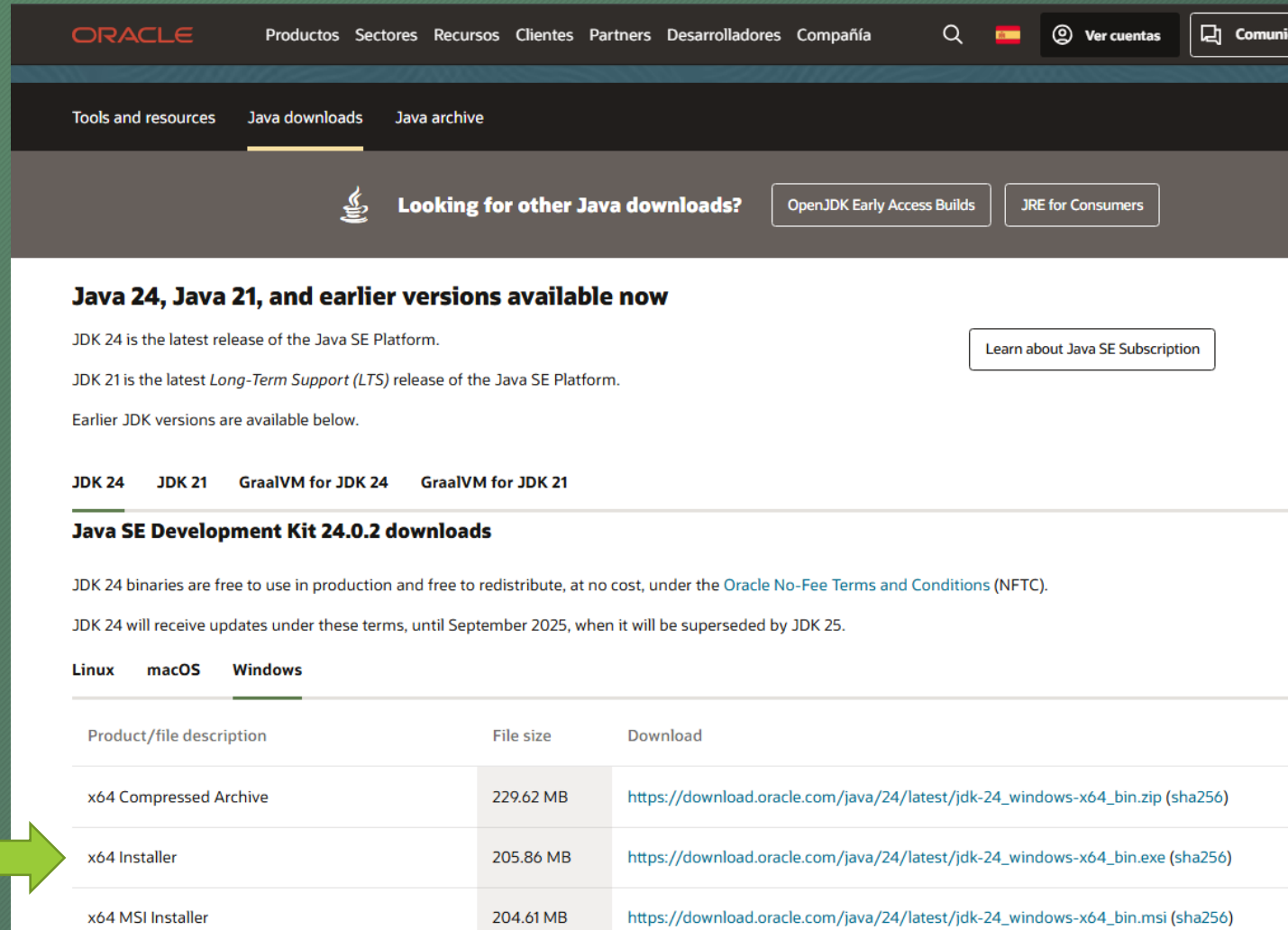
IDE més emprats per programar en Java:





Eclipse, NetBeans, IntelliJ Idea, BlueJ...

Eclipse


Instal·lacions

Instal·lar JAVA: <https://www.oracle.com/es/java/technologies/downloads/#jdk24-windows>



ORACLE Productos Sectores Recursos Clientes Partners Desarrolladores Compañía    Ver cuentas  Comuni

Tools and resources **Java downloads** Java archive

 **Looking for other Java downloads?** [OpenJDK Early Access Builds](#) [JRE for Consumers](#)

Java 24, Java 21, and earlier versions available now

JDK 24 is the latest release of the Java SE Platform. [Learn about Java SE Subscription](#)

JDK 21 is the latest *Long-Term Support (LTS)* release of the Java SE Platform.

Earlier JDK versions are available below.

JDK 24 **JDK 21** **GraalVM for JDK 24** **GraalVM for JDK 21**

Java SE Development Kit 24.0.2 downloads

JDK 24 binaries are free to use in production and free to redistribute, at no cost, under the [Oracle No-Fee Terms and Conditions \(NFTC\)](#).

JDK 24 will receive updates under these terms, until September 2025, when it will be superseded by JDK 25.

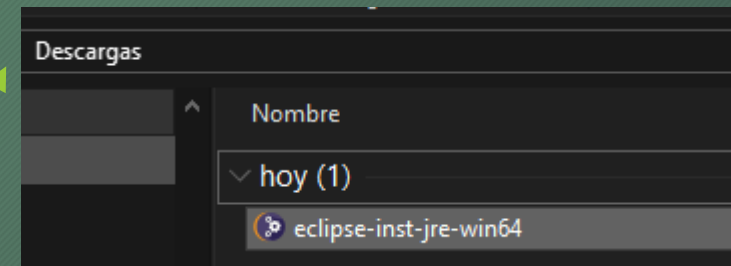
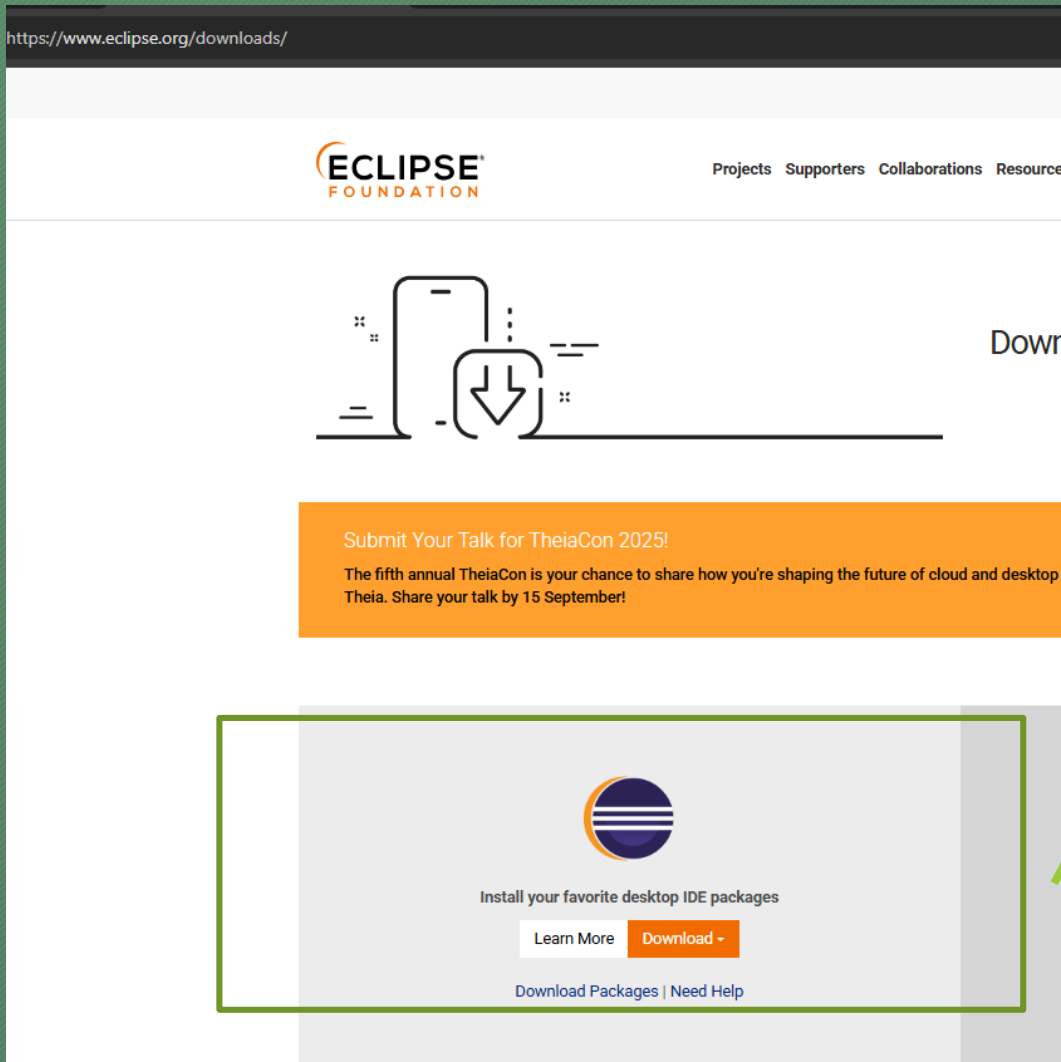
Linux **macOS** **Windows**

Product/file description	File size	Download
x64 Compressed Archive	229.62 MB	https://download.oracle.com/java/24/latest/jdk-24_windows-x64_bin.zip (sha256)
x64 Installer	205.86 MB	https://download.oracle.com/java/24/latest/jdk-24_windows-x64_bin.exe (sha256)
x64 MSI Installer	204.61 MB	https://download.oracle.com/java/24/latest/jdk-24_windows-x64_bin.msi (sha256)

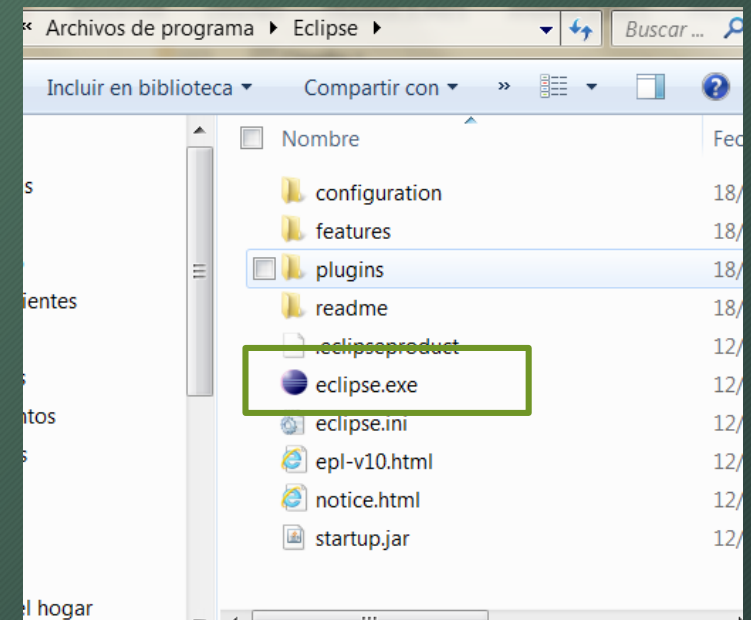
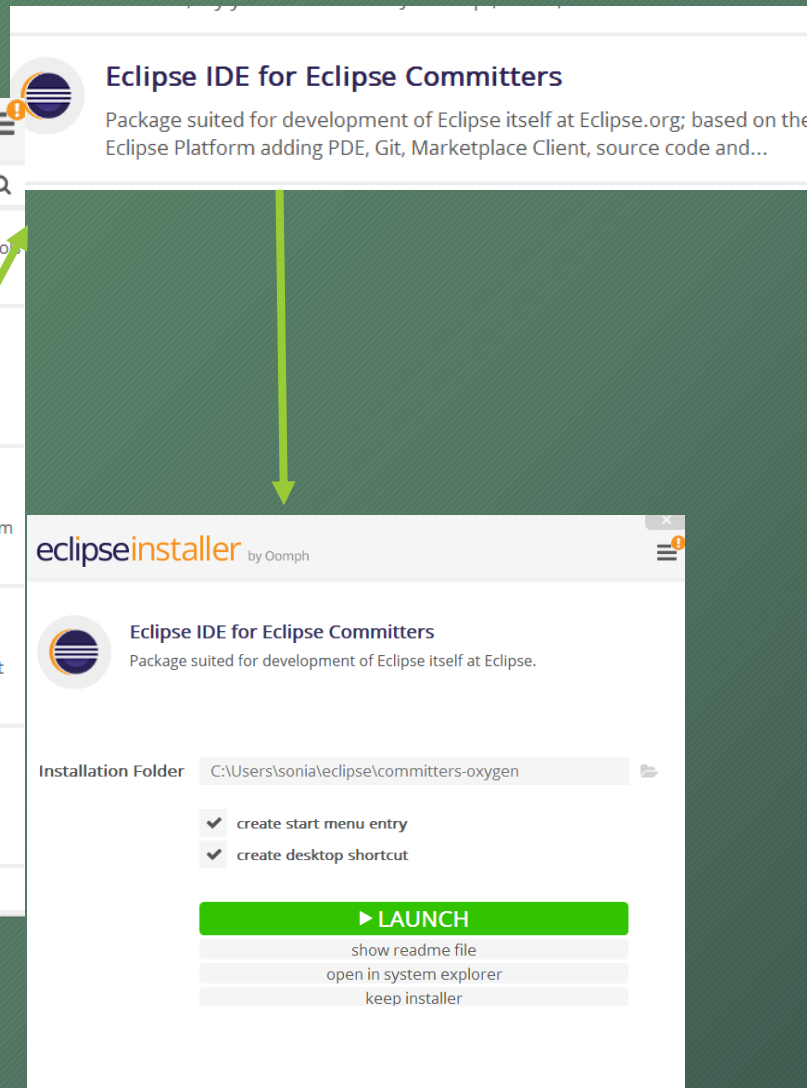
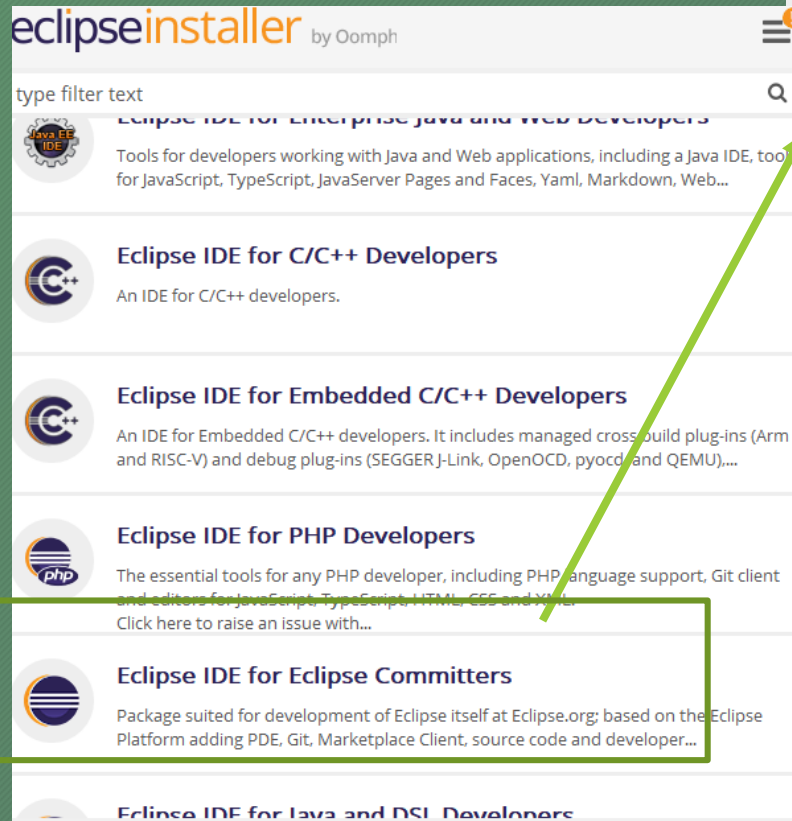
Instal·lacions

Instal·lar Eclipse: <https://www.eclipse.org/downloads/>

Descomprimir o instal·lar en la unitat Principal.

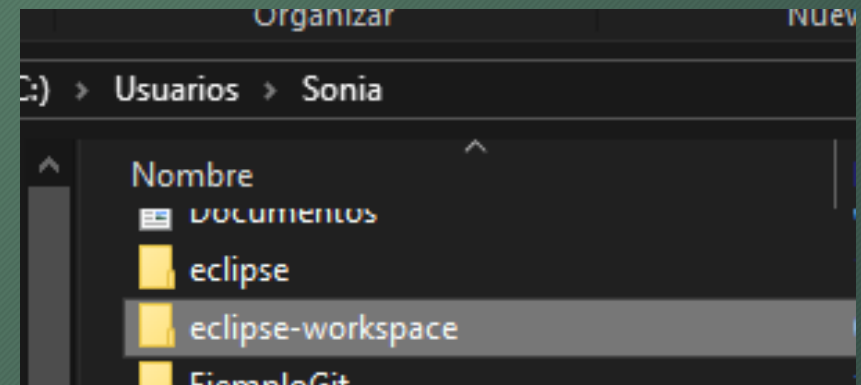
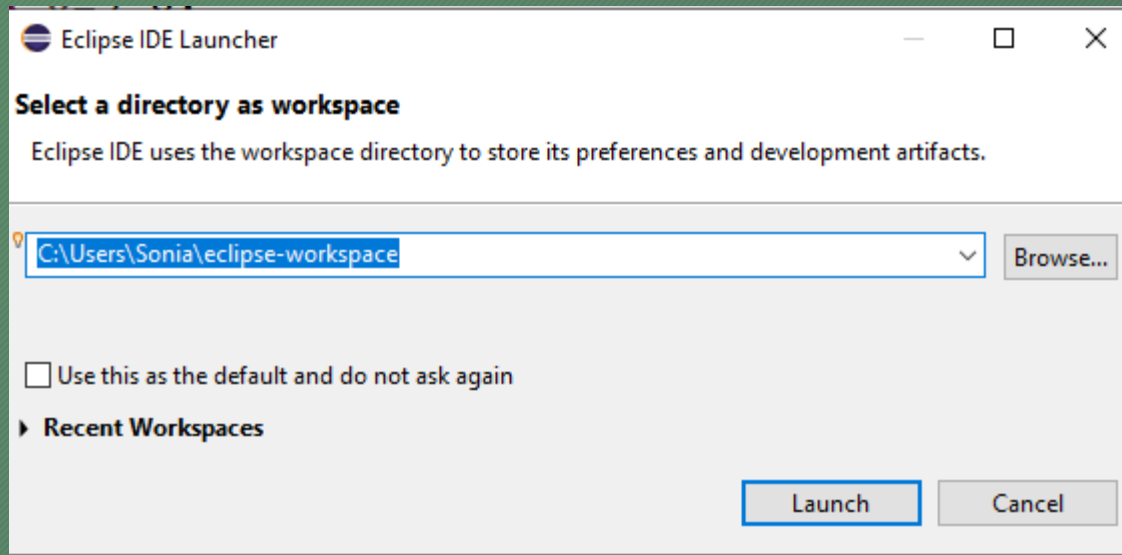


Com instal·lar eclipse: <https://jarroba.com/instalar-bien-eclipse-un-ide-de-muchos/>
Des del propi instal·lador es pot escollir després qualsevol altre entorn.



Com instal·lar eclipse: <https://jarroba.com/instalar-bien-eclipse-un-ide-de-muchos/>
Des del propi instal·lador es pot escollir després qualsevol altre entorn.

Es pot definir la ubicació de l'espai de treball (workspace)





Eclipse IDE for Eclipse Committers

Package Description

Package suited for development of Eclipse itself at Eclipse.org; based on the Eclipse Platform adding PDE, Git, Marketplace Client, source code and developer documentation.

Click [here](#) to raise an issue with the Eclipse Platform.

Click [here](#) to open a bug report with the Eclipse Git team provider.

This package includes:

- Git integration for Eclipse
 - Eclipse Java Development Tools
 - Maven Integration for Eclipse
 - Eclipse Plug-in Development Environment
- Detailed features list

<https://www.eclipse.org/downloads/packages/release/2023-03/r/eclipse-ide-eclipse-committers>

Millors IDE per programar :<https://www.diarlu.com/mejores-ide-programar-java/>

Tipus programa Java

- Aplicacions de consola (Shell)
- Aplicacions de propòsit en general
- Miniaplicacions (programes que s'executen en el navegador)

Nomenclatura :

<https://jcodepoint.com/java/convencion-de-nombres-en-java/>

<https://www.discoduroderoer.es/convencion-de-nombres-en-java/>

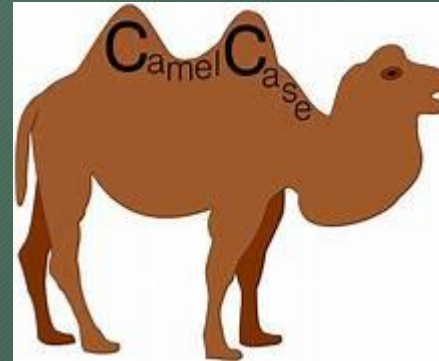
Tipus de notació:

<https://www.neoguias.com/tipos-notacion-nombres/>

<https://www.aluracursos.com/blog/convenciones-de-nomenclatura-camel-pascal-kebab-snake-case>

Existeixen dos tipus de camelCase:

- **UpperCamelCase** (més conegut com PascalCase), quan la primera lletra de cadascuna de les paraules és majúscula. Exemple: ExempleDeUpperCamelCase.
- **lowerCamelCase** (o simplement camelCase), igual que l'anterior amb l'excepció que la primera lletra és minúscula. Exemple: exempleDeLowerCamelCase.



En Java:

- **Paquets** : noms de paquets amb un nom de domini en minúscules, seguit de components addicionals separats per punts. **exemple.paquet.nom**
- **Classes**: substantius , noms de classes descriptius i significatius i seguir el format **PascalCase**. Per una paraula : Cotxe per més d'una: ExempleClass
- **Interfícies**: descriptius i utilitzar adjectius o substantius per a indicar el seu propòsit, es recomana seguir el format **PascalCase** on la primera lletra de cada paraula està en majúscula. ExempleInterficiNom
- **Variables**: **camelCase**, per exemple :userNom. No poden començar per números. Si és una paraula int numero, si són dos també es poden posar int numEntero o int num_entero.
- **Mètodes**: utilitzant verbs o frases verbals que descriguin amb precisió l'acció que realitzen dins del codi, per exemple: calcularPreuTotal. **camelCase**.
- **Constants**: es declaren usant lletres majúscules i guions baixos. Final int MAX_INTENTS=3
- **Nom projecte**: **camelCase**, per exemple: primerProjecte

En Java, cada programa comença amb una **classe principal**.

La declaració d'aquesta classe és essencial, és la classe que actua com el punt d'inici de l'execució d'un programa.

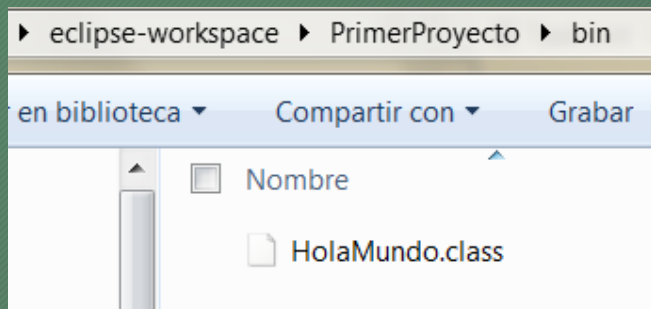
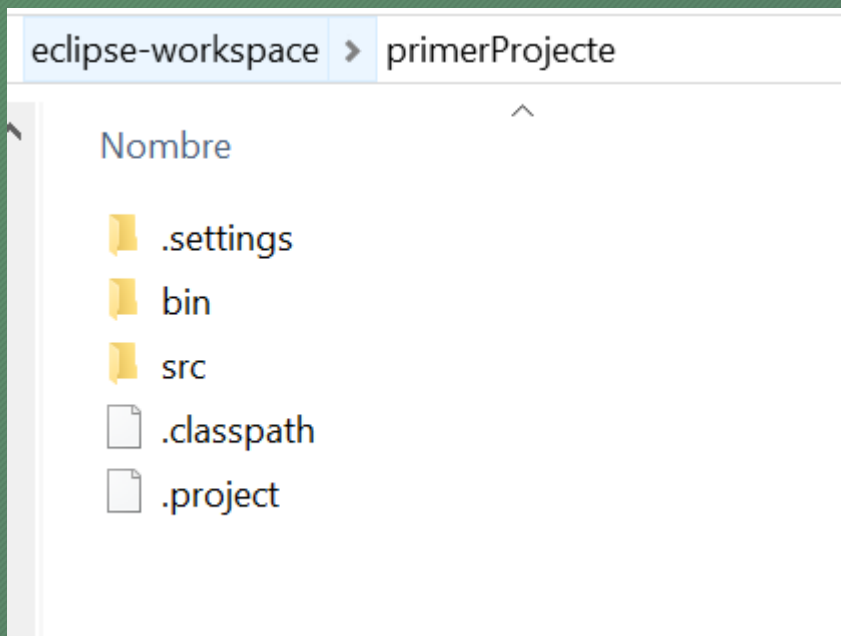
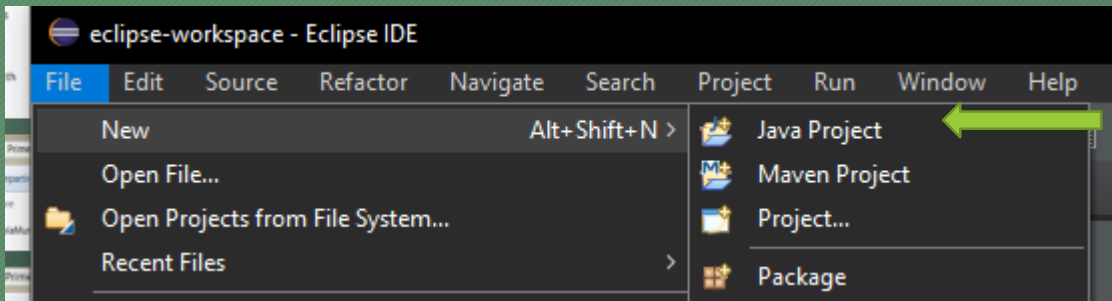
Aquesta classe ha de tenir el mètode main:

```
public static void main(String[] args) {  
    // Código del programa  
}
```

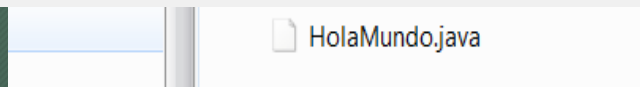
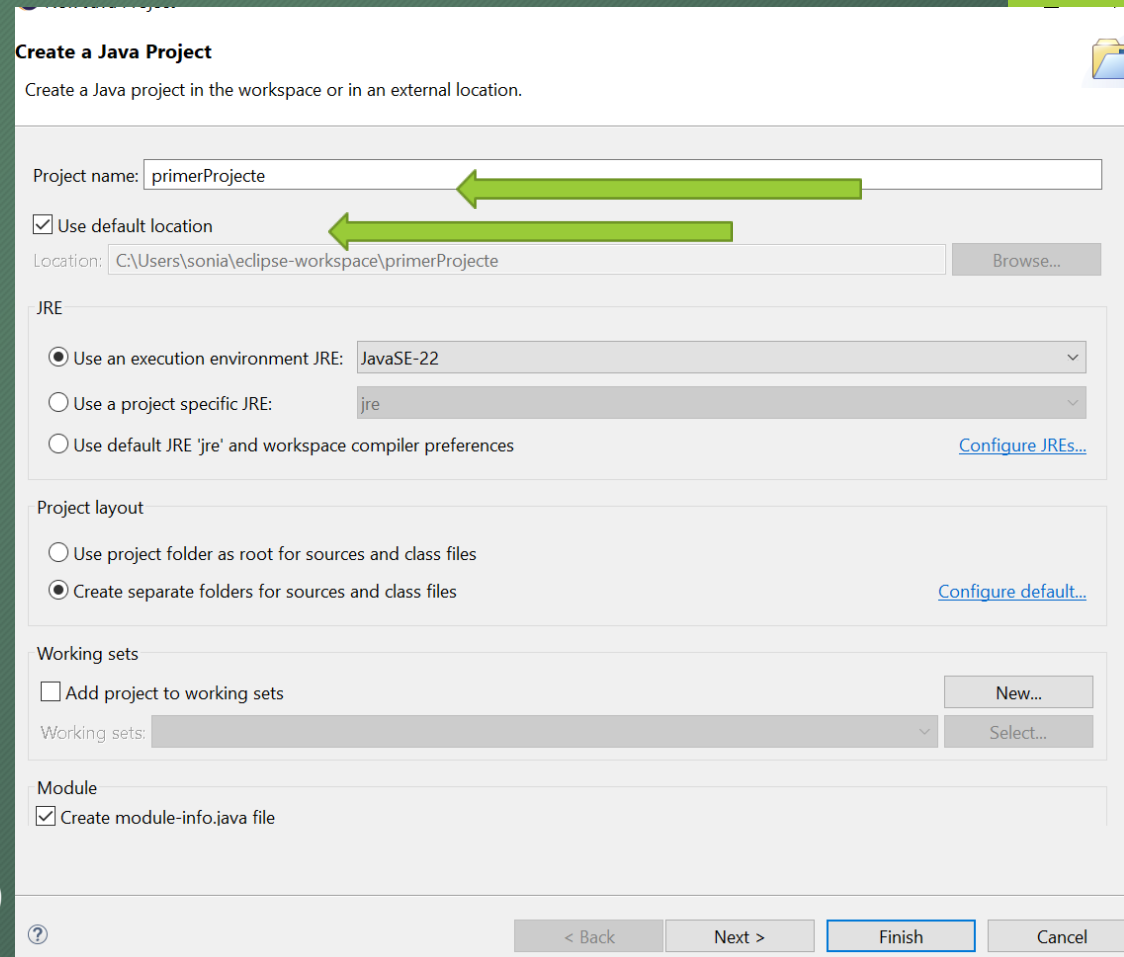
Mètode main, public i static.
Visible i sense necessitat de crear una
Instància.

```
*/  
public class HolaMundo {  
    public static void main(String[] args) {  
  
    }  
}
```

Classe principal, HolaMundo



Classes pròpies i
predefinides (api)

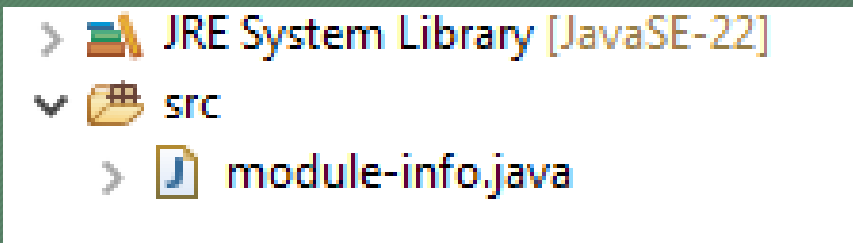
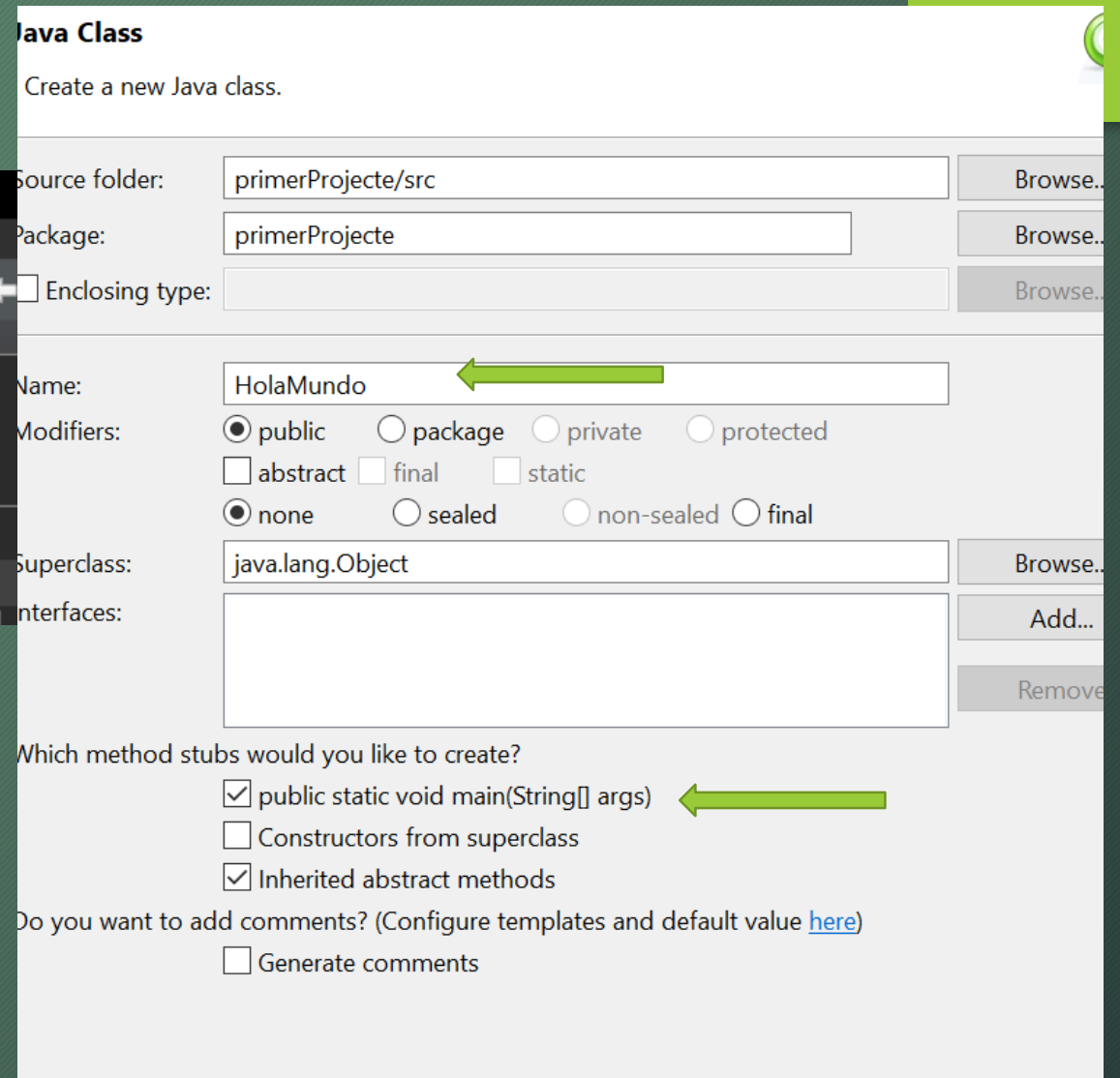
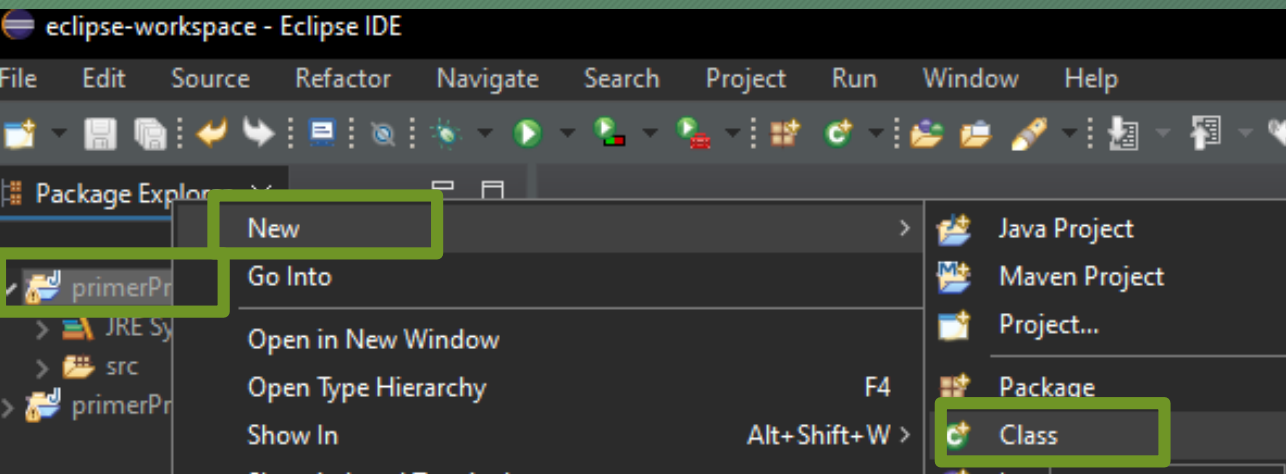


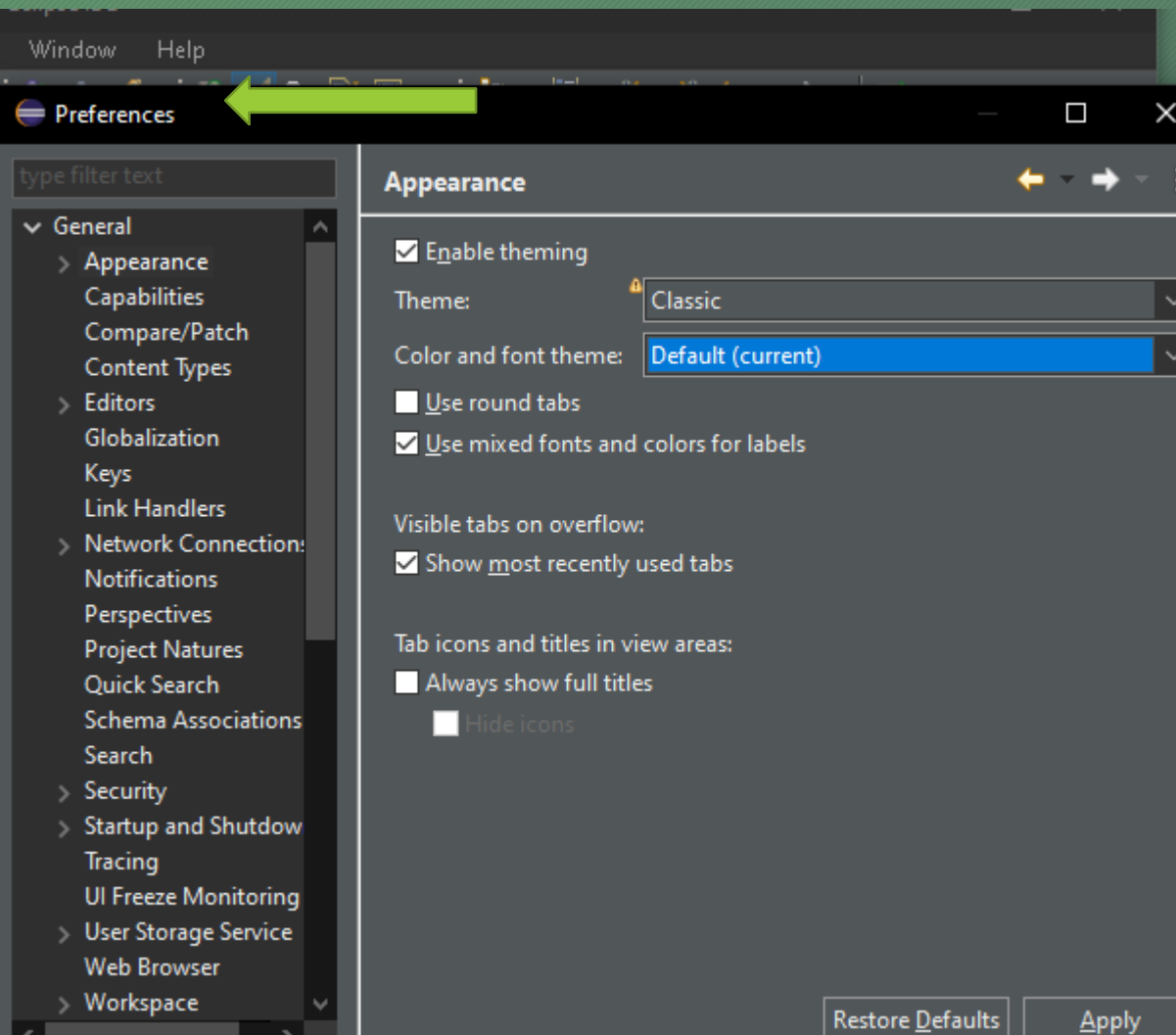
Aplicacions de consola

Tot projecte ha de tenir almenys una classe, anem a crear una classe principal.

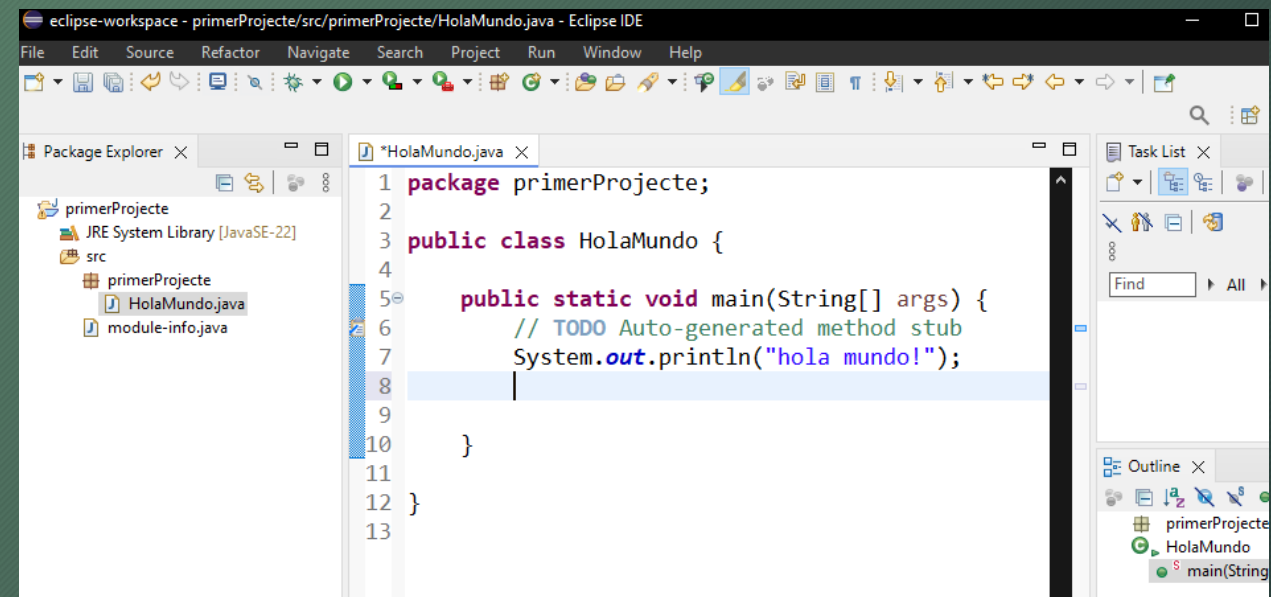
Estructura mínima de una classe (main)

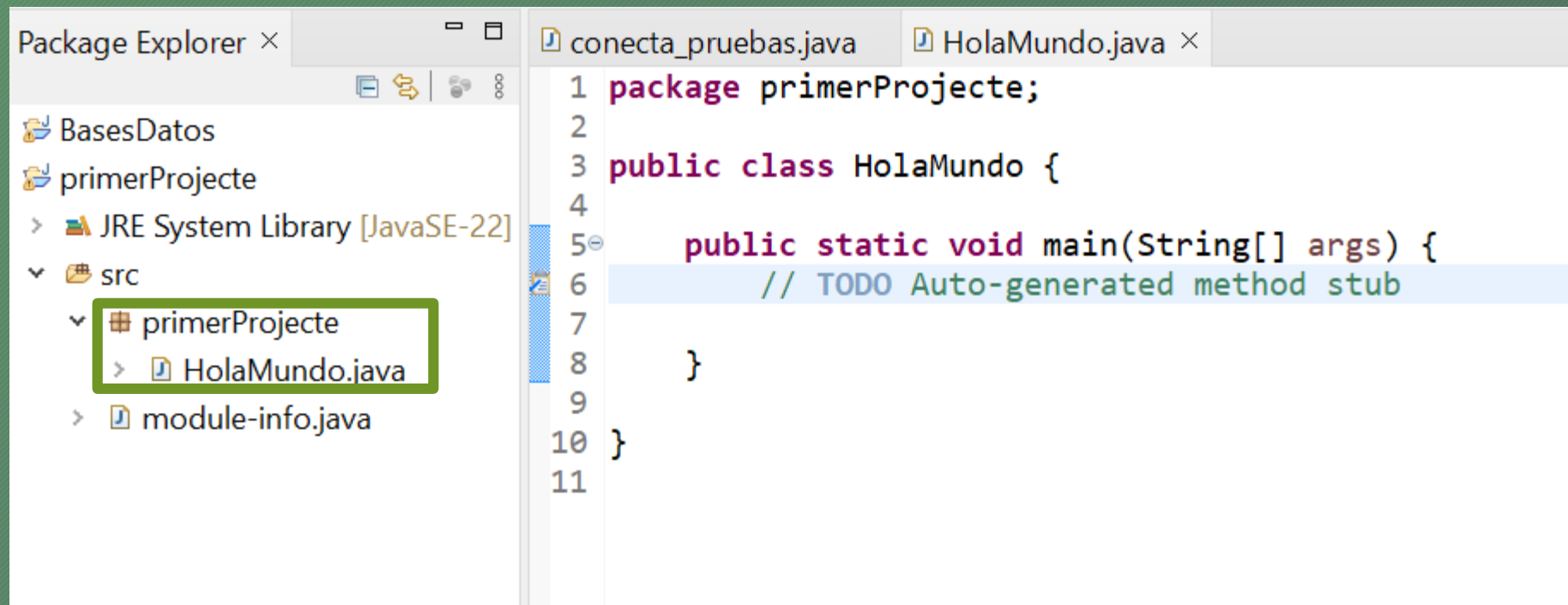
consola



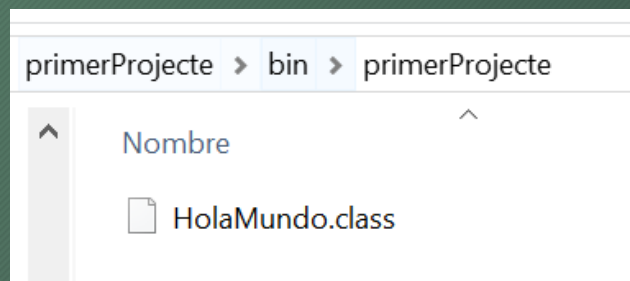
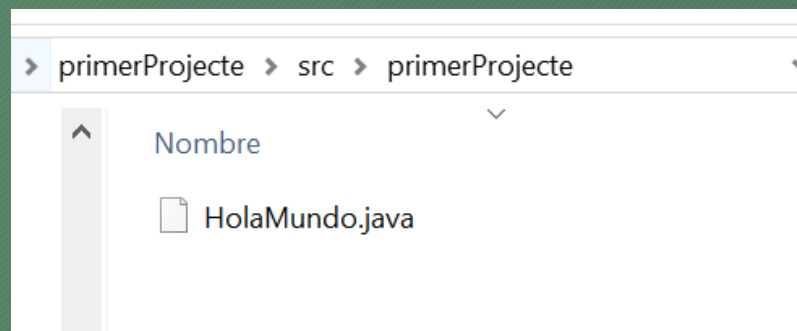


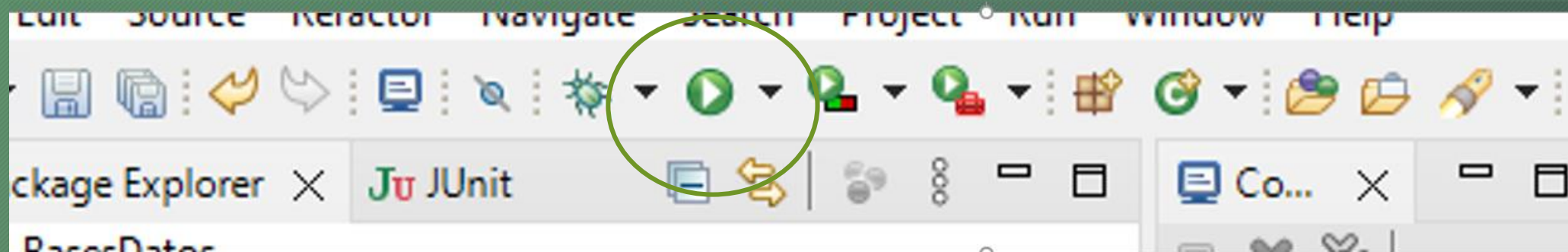
Windows/Preferences/General/Appearance/Theme



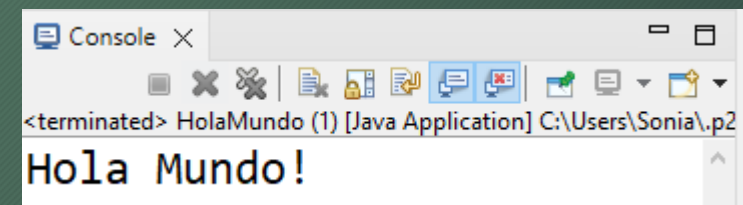


consola





```
package primerProyecto;  
  
public class HolaMundo {  
  
    public static void main(String[] args) {  
        // TODO Auto-generated method stub  
        System.out.println("Hola Mundo!");  
    }  
}
```




```
6  
7 int edad=40;
```

Warning

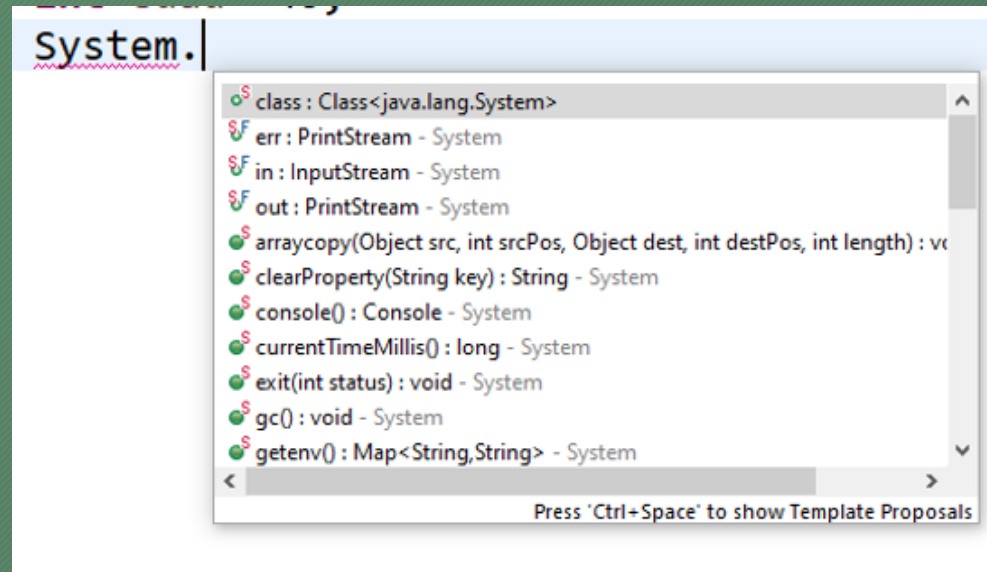
```
7 int edad=42;  
8 Sytem.out.println("Mi edad es: "+edad);  
9
```

Error

consola

```
1 package segundoProyecto;  
2  
3 public class HolaMundo {  
4  
5     public static void main(String[] args) {  
6         // TODO Auto-generated method stub  
7         int edad =40;  
8         Sytem.out.println("Mi edad es :"+edad);  
9  
10  
11     }  
12  
13 }  
14
```

Intellisense



consola

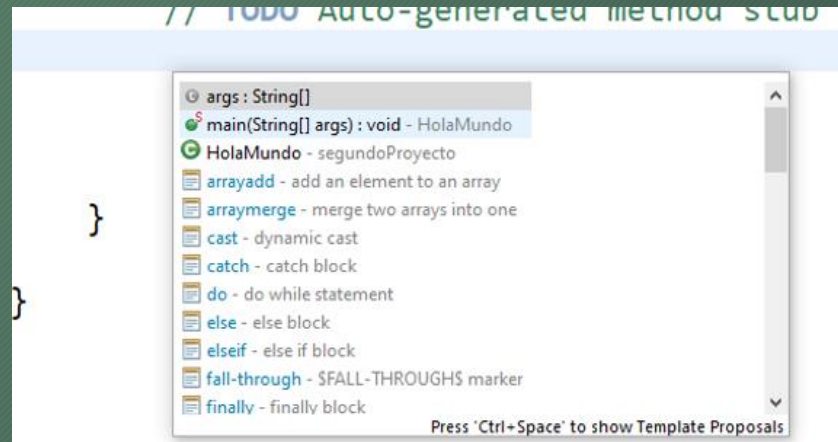
Abreviacions

syso o sysout, seguit de Ctrl + Barra
espai (space)

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

Template proposals (Propuestas de plantilla)

Ctrl + Space



Paraules reservades i
variables definides