

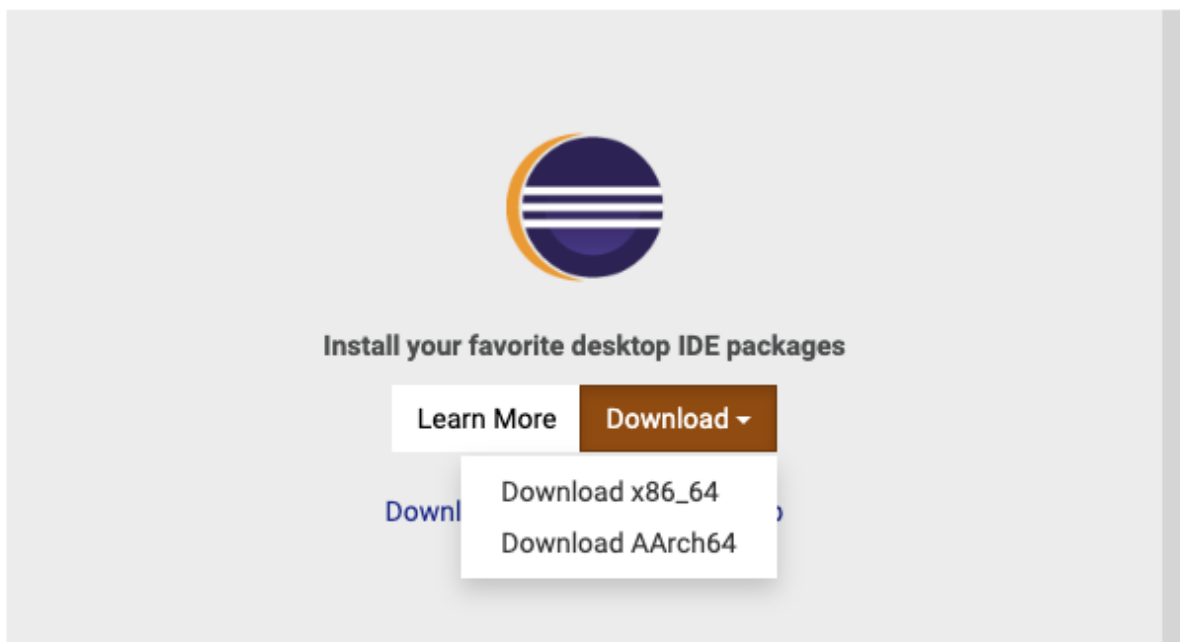
# TAREA 1 DE PROGRAMACIÓN

En mi caso estoy usando el sistema operativo Mac por tanto los pasos de instalación los voy a realizar sobre este sistema operativo.

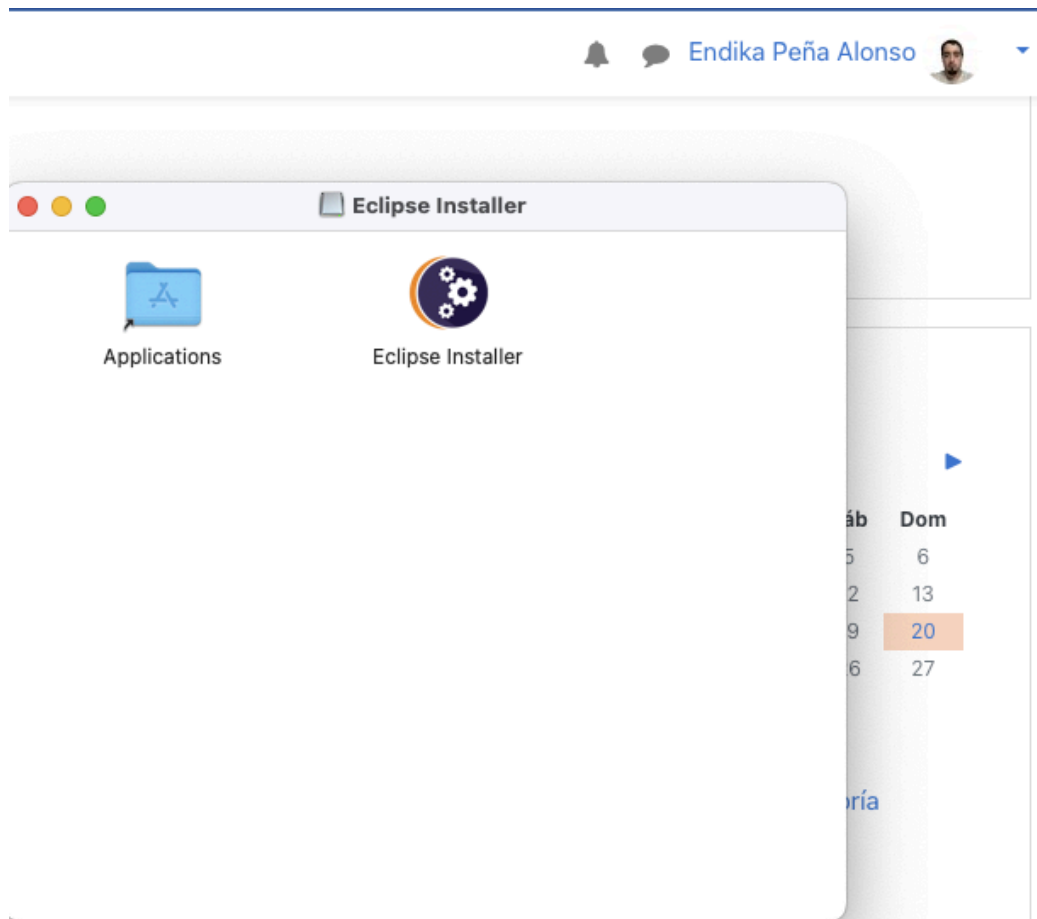
<b>Instalación de eclipse</b>	<b>1</b>
Iniciando el proyecto y ejecución	6
<b>Instalación de IntelliJ</b>	<b>9</b>
Iniciando el proyecto y ejecución	11

## Instalación de eclipse

Debemos descargar eclipse primero de su página oficial <https://www.eclipse.org/downloads/> en mi caso la versión AArch64 puesto que mi equipo cuenta con un chip ARM

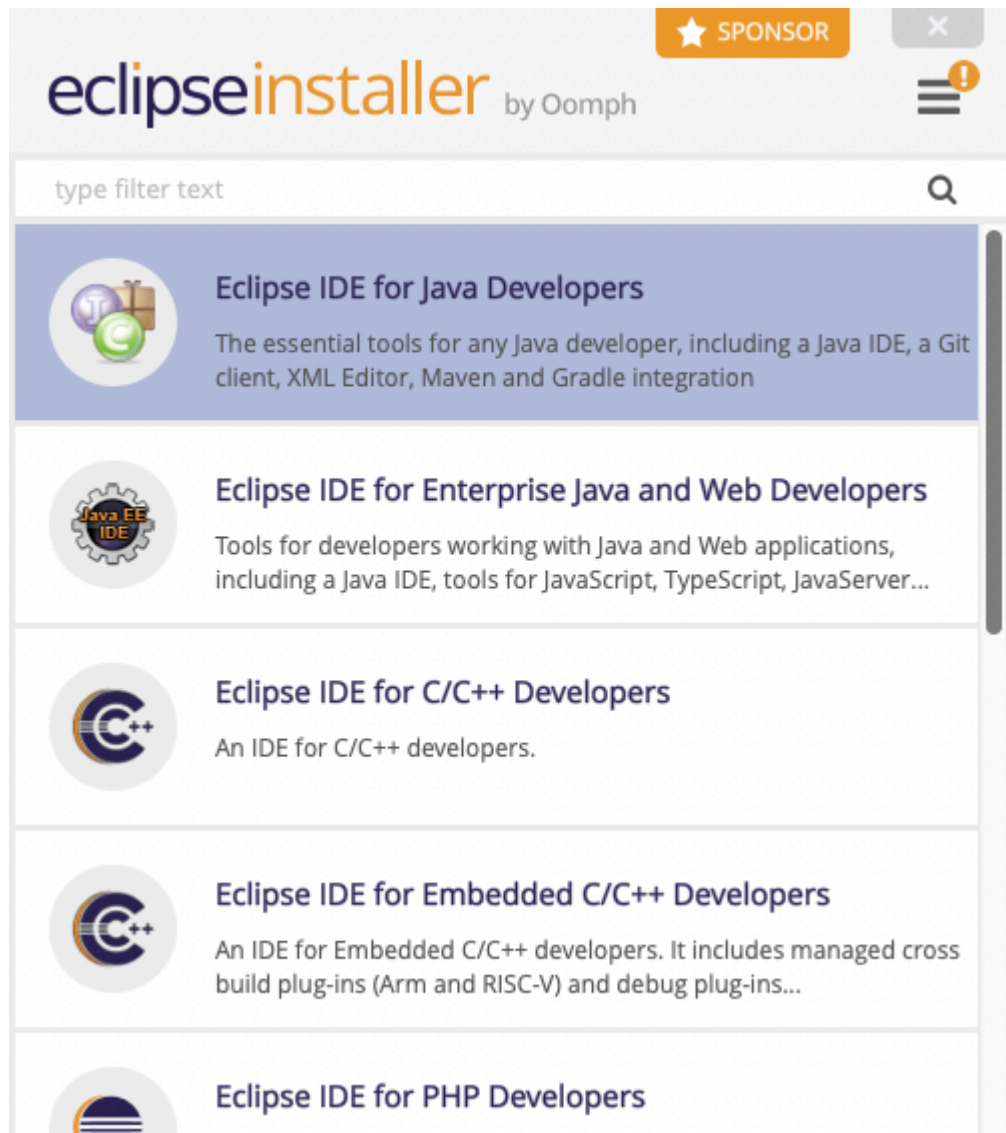


Una vez descargado en mi caso es un archivo .DMG al abrirlo nos aparecerá un icono del programa y un enlace simbólico a la carpeta Applications debemos arrastrar el icono allí para copiar el instalador como tal.



Una vez realizado este paso debemos ejecutar el Eclipse Installer y permitir en mac su ejecución ya que el software descargado desde fuentes ajenas a la App Store no tiene las firmas de confianza.

Una vez arrancado aparece está ventana



Debemos marcar que queremos instalar Eclipse para desarrolladores JAVA

Cuando pulsemos esto nos preguntará donde queremos instalarlo dejará en esa ubicación los archivos del programa.



## Eclipse IDE for Java Developers

[details](#)

The essential tools for any Java developer, including a Java IDE, a Git client, XML Editor, Maven and Gradle integration.

Java 21+ VM

JRE 23.0.0 - <https://download.eclipse.org/jre/>



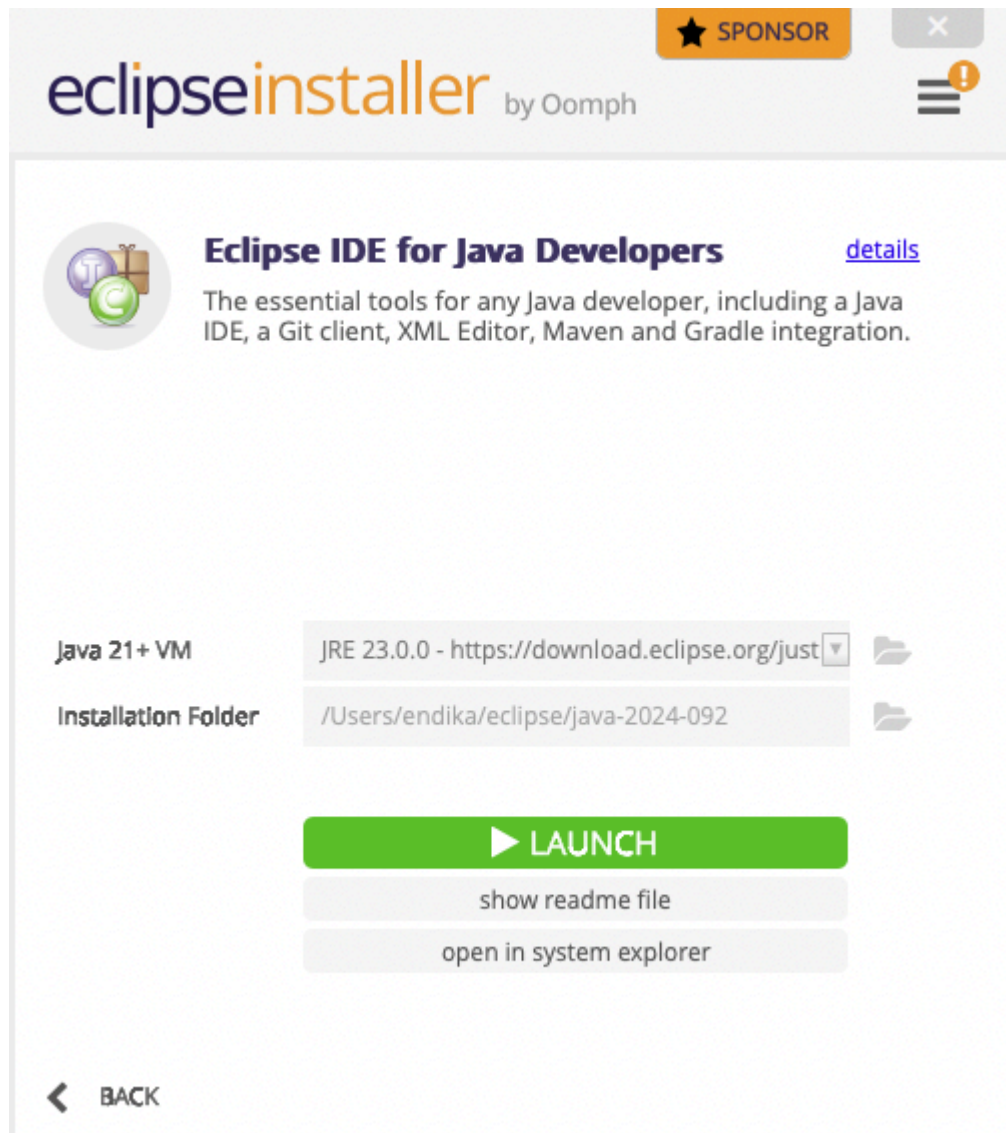
Installation Folder

/Users/endika/eclipse/java-2024-092

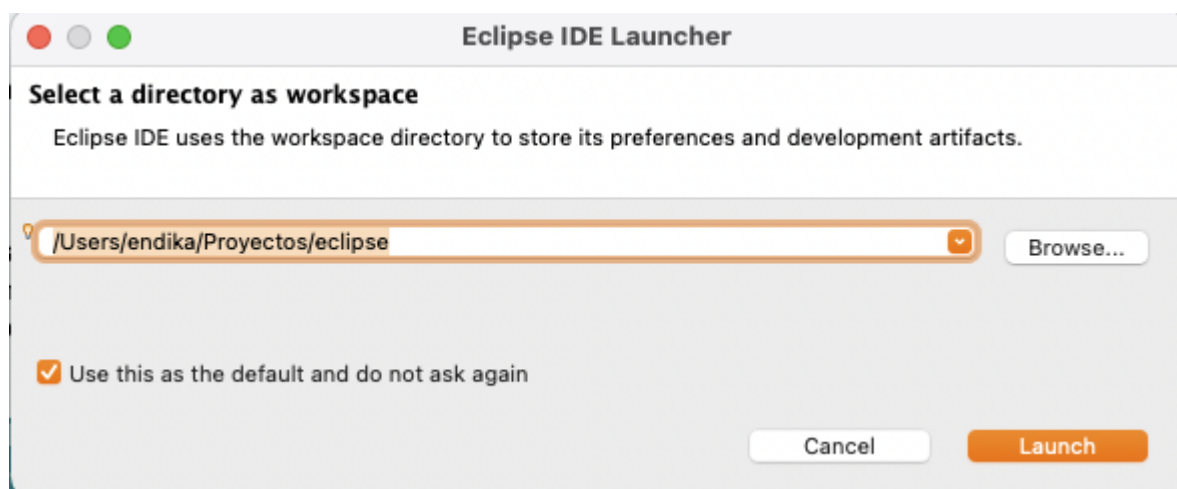


 **INSTALL**

 **BACK**



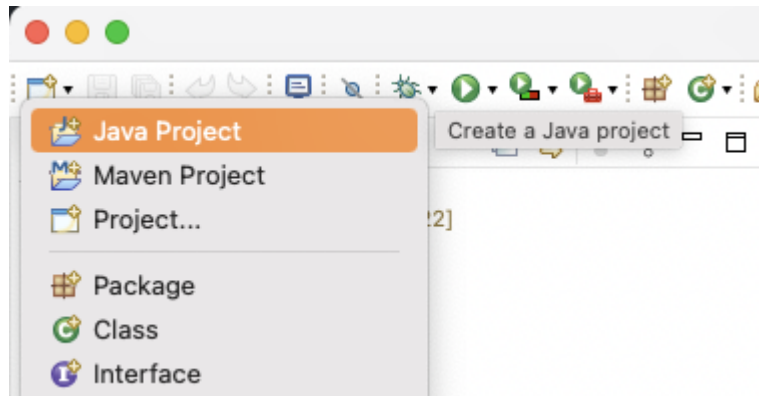
Al iniciar eclipse nos preguntará donde queremos nuestro espacio de trabajo.



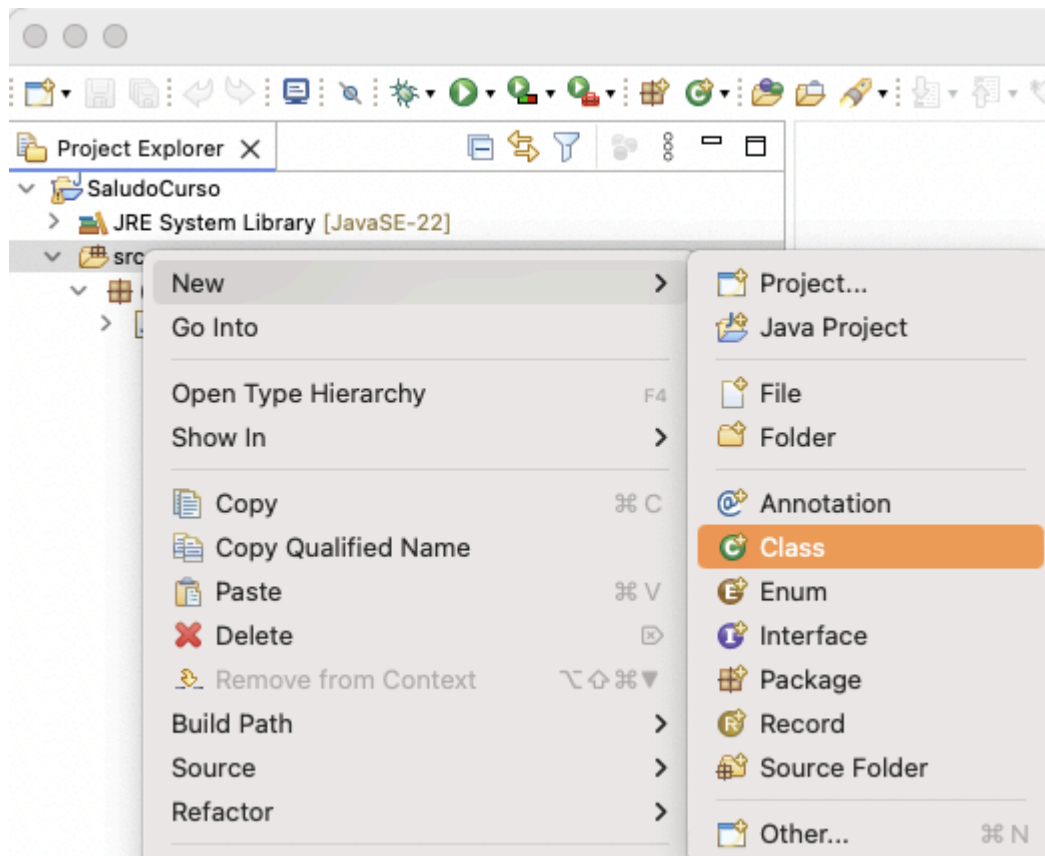
Ya podemos crear proyectos JAVA.

## Iniciando el proyecto y ejecución

Lo primero debemos crear un proyecto Java para ello margen superior izquierdo y java project o seguimos el Wizard



Una vez creado el proyecto pasamos a crear nuestra primera clase java



**New Java Class**

**Java Class**

The use of the default package is discouraged.

Source folder:

Package:  (default)

☐ Enclosing type:

---

Name:

Modifiers:

☒ public ☐ package ☐ private ☐ protected

☐ abstract ☐ final ☐ static

☒ none ☐ sealed ☐ non-sealed ☐ final

Superclass:

Interfaces:

Which method stubs would you like to create?

☐ public static void main(String[] args)

☐ Constructors from superclass

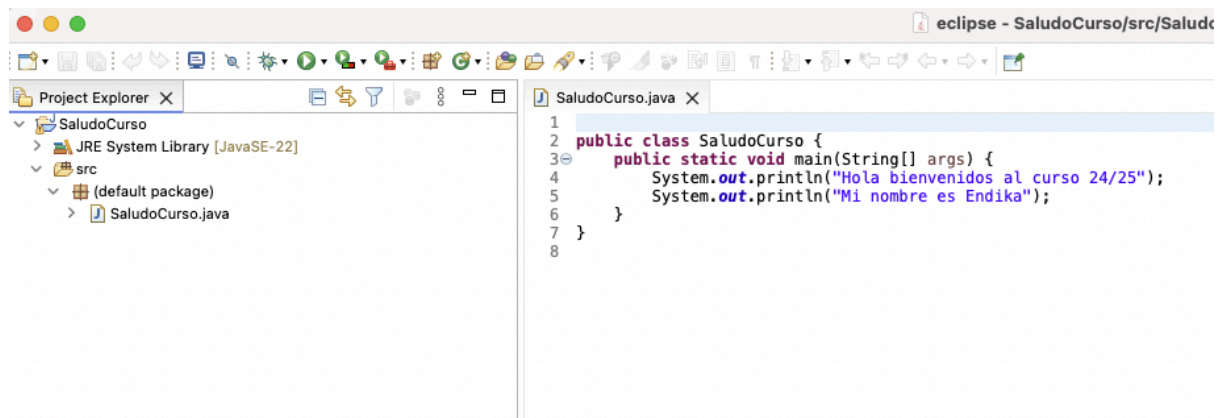
☒ Inherited abstract methods

Do you want to add comments? (Configure templates and default value [here](#))

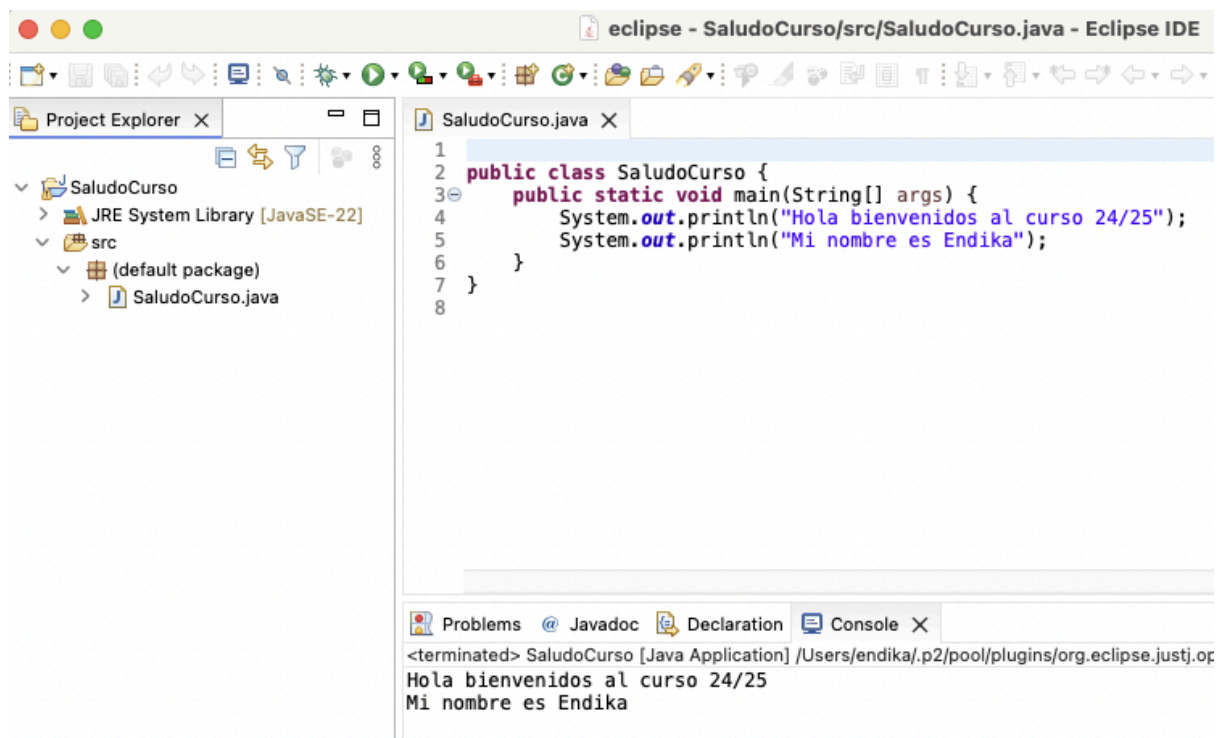
☐ Generate comments

Abrimos el fichero SaludoCurso.java y pegamos el contenido que nos han pasado en el enunciado.

```
public class SaludoCurso {  
    public static void main(String[] args) {  
        System.out.println("Hola bienvenidos al curso 24/25");  
        System.out.println("Mi nombre es TU NOMBRE");  
    }  
}
```



Una vez copiado y modificado el nombre. Pulsamos el botón superior verde de play para ejecutar el programa.



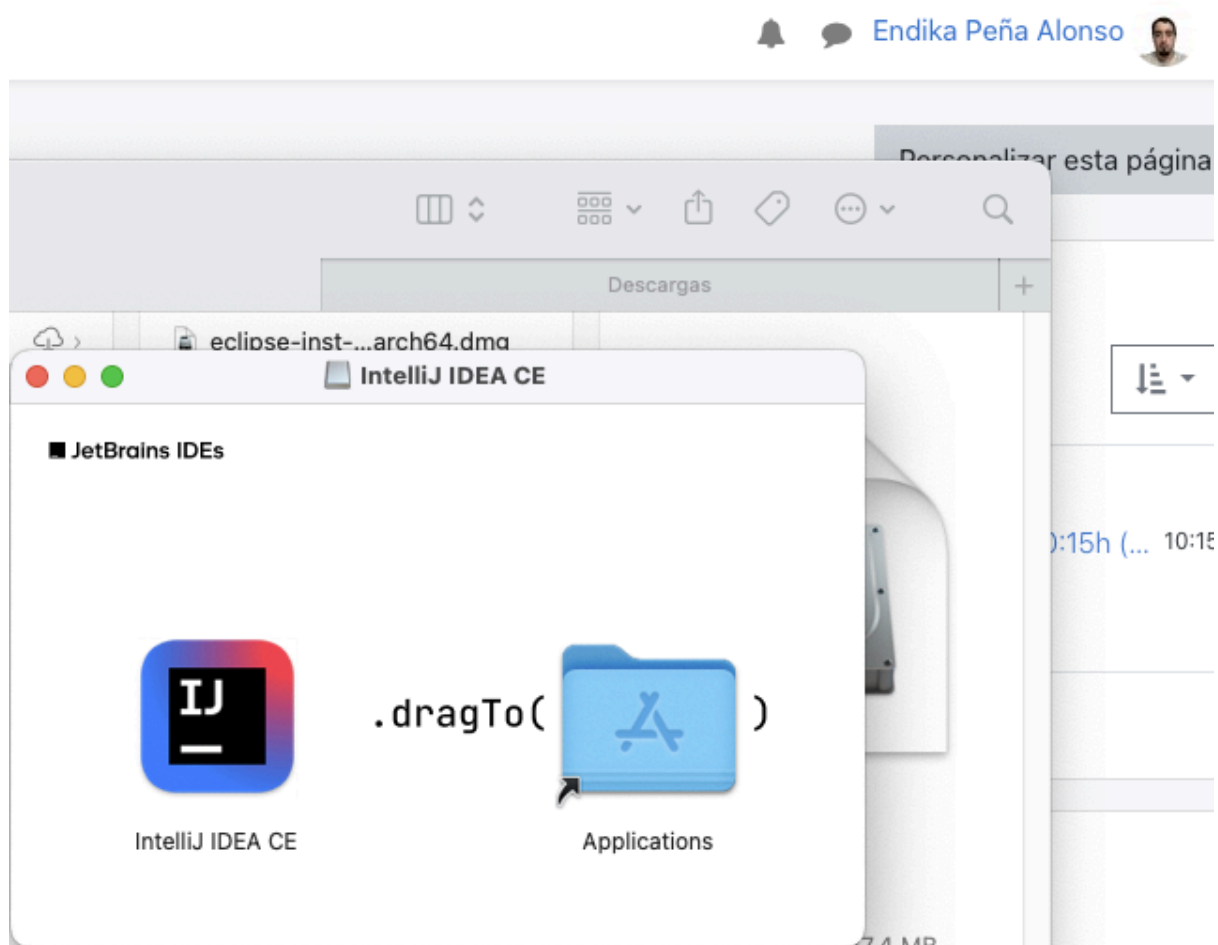


# Instalación de IntelliJ

Para instalar IntelliJ primero debemos descargar el software previamente desde su página oficial <https://www.jetbrains.com/es-es/idea/download/> en mi caso descargaré la versión para mac. Puesto que no voy a comprar el software voy a descargar la versión Community Edition que es gratuita.

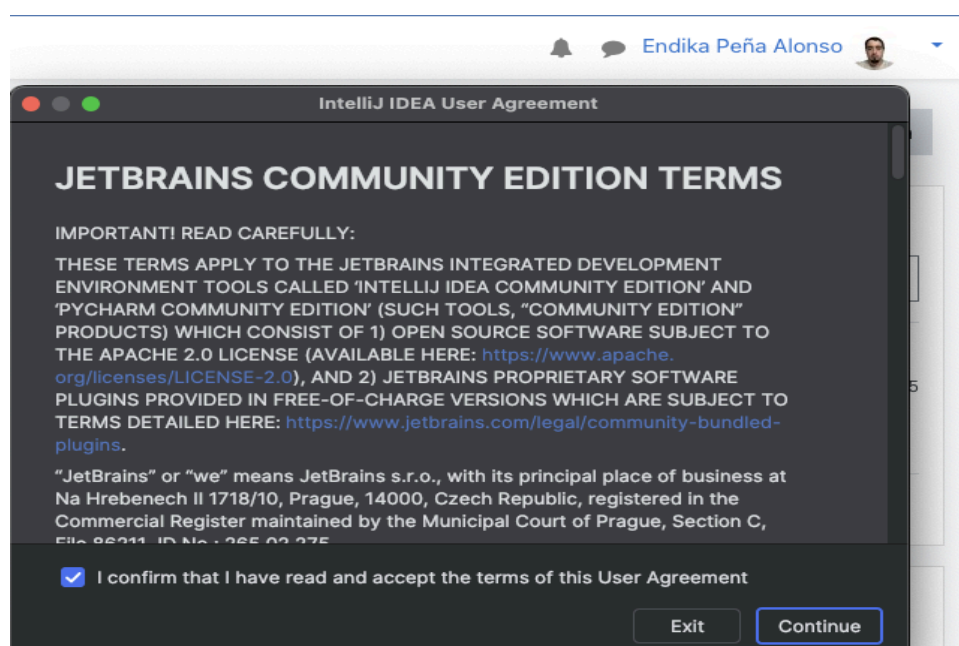


El proceso de instalación es similar al de Eclipse abrimos el DMG de apple y arrastramos el icono hasta el enlace simbólico de la carpeta applications



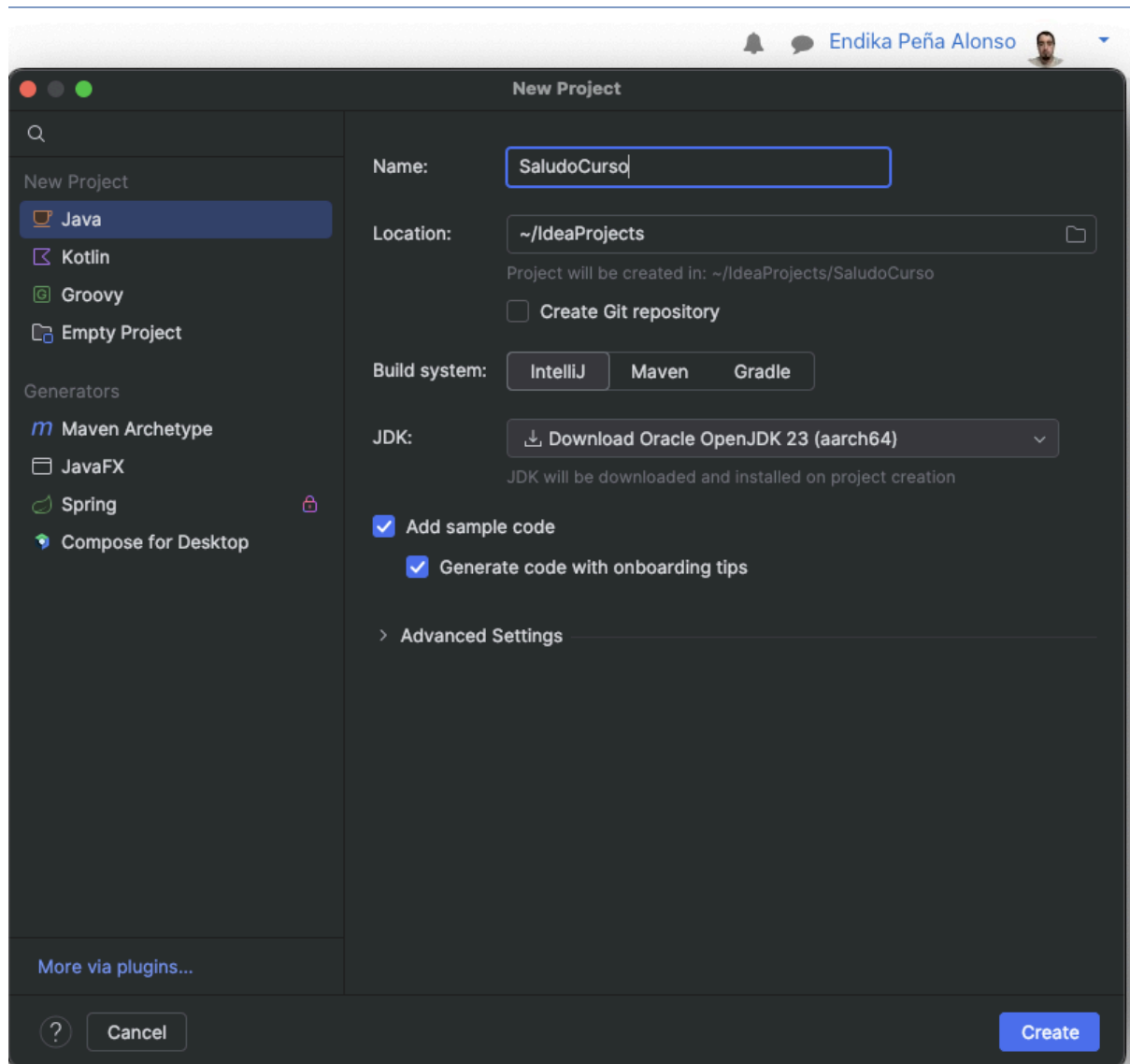
Una vez realizado esto iniciamos la aplicación de IntelliJ Idea usando el launchpad de apple (como el botón de inicio en windows) para buscar nuestra aplicación de IDE.

Nos pedirá que aceptemos los términos de uso y condiciones de la versión community

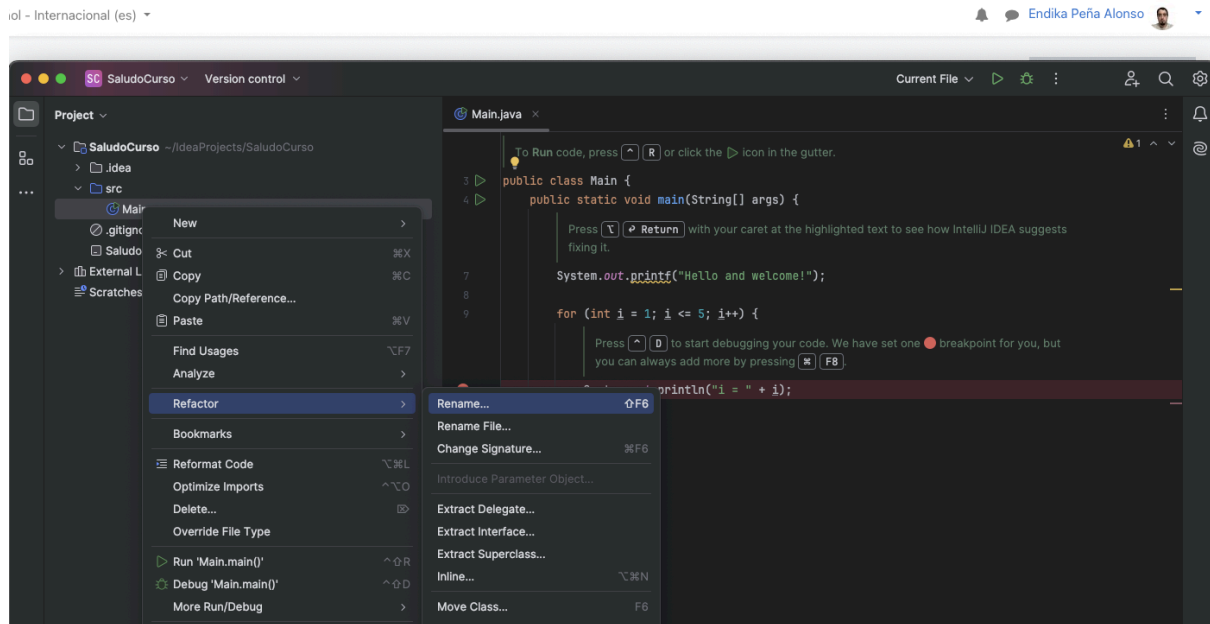


## Iniciando el proyecto y ejecución

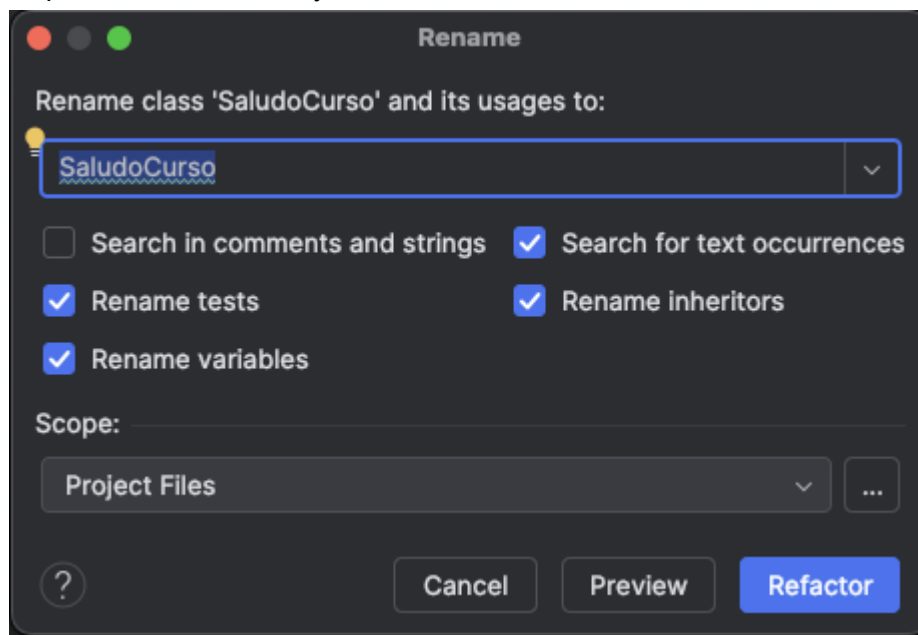
En la siguiente ventana aparecen unos enlaces rápidos para crear nuestro proyecto, rellenamos y pulsamos en create.



Una vez realizado esto tenemos nuestro proyecto en el que vamos a renombrar el fichero Main que hay en el src para que se llame como queremos SaludoCurso

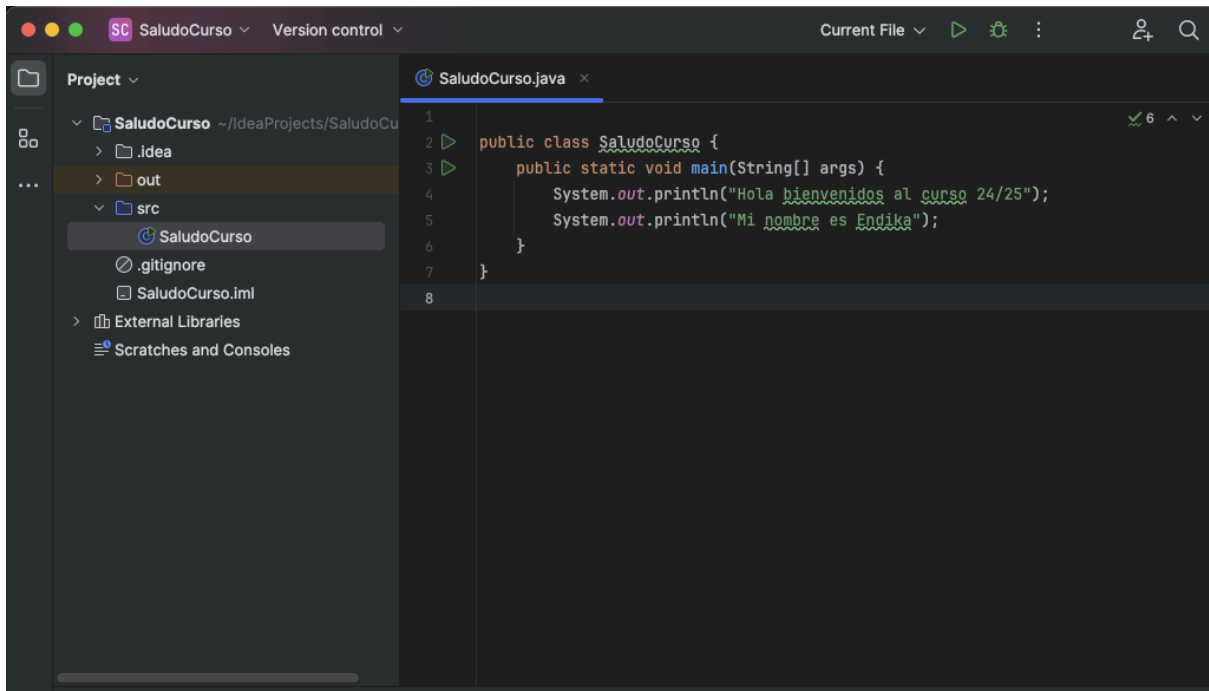


Le ponemos el nombre y le damos a refactor



Ahora pegamos el código que nos habían dado en el enunciado.

```
public class SaludoCurso {
    public static void main(String[] args) {
        System.out.println("Hola bienvenidos al curso 24/25");
        System.out.println("Mi nombre es TU NOMBRE");
    }
}
```



Y pulsando el botón verde que hay cerca de la esquina superior derecha de la imagen, podemos compilar y ejecutar el código.

