



**TÉCNICO SUPERIOR EN DESARROLLO DE
APLICACIONES MULTIPLATAFORMA**

DEPARTAMENTO DE INFORMÁTICA

EZJAVACODE

MANUAL DE USUARIO

RUBÉN MATAMOROS TRIGO
2024/2025

ÍNDICE

| | |
|--|-----------|
| Introducción | 3 |
| 1. CREAR CLASE | 4 |
| Menú Principal. | 4 |
| Menú Principal - Botón Crear Clase | 4 |
| Crear Clase - Nombre de la clase | 5 |
| 1.1 CREAR CLASE – CREAR ATRIBUTO..... | 6 |
| Crear Clase - Crear Atributos 1 | 6 |
| Crear Clase - Crear Atributos 2 | 6 |
| Crear Clase - Crear Atributos 3 | 7 |
| Crear Clase - Crear Atributos 4 | 7 |
| Crear Clase - Crear Atributos 5 | 8 |
| Crear Clase - Crear Atributos 6 | 8 |
| 1.2 CREAR CLASE – CREAR METODO | 9 |
| Crear Clase - Crear Métodos 1 | 9 |
| Crear Clase - Crear Métodos 2..... | 9 |
| Crear Clase - Crear Métodos 3..... | 10 |
| Crear Clase - Crear Métodos 4..... | 10 |
| Crear Clase - Crear Métodos 5..... | 10 |
| Crear Clase - Crear Métodos 6..... | 11 |
| Crear Clase - Crear Métodos 7 | 11 |
| Crear Clase - Crear Métodos 8..... | 11 |
| Crear Clase - Crear Métodos 9..... | 11 |
| Crear Clase - Crear Métodos 10..... | 12 |
| Crear Clase - Crear Métodos 11..... | 12 |
| Crear Clase - Crear Métodos 12..... | 12 |
| 1.3 CREAR CLASE – PREVISUALIZACIÓN..... | 13 |
| Crear Clase - previsualización | 13 |
| 2. MIS CLASES | 14 |
| Menú Principal - Botón Mis clases..... | 14 |
| Mis Clases - Pantalla Principal | 14 |
| Mis Clases - Botón editar | 14 |
| Mis Clases - Botón Visualizar | 15 |
| Mis Clases - Botón Eliminar | 15 |
| 3. AJUSTES..... | 16 |
| Menú Principal - Botón Ajustes..... | 16 |
| Ajustes - Pantalla principal..... | 16 |
| Ajustes - Botón Elegir carpeta | 17 |
| Ajustes - Botón Ruta por defecto | 17 |
| 4. LIBRERIAS..... | 18 |
| Menú Principal - Botón Librerías..... | 18 |
| Librerías - Mi librería de Funciones | 18 |

Introducción

El presente manual tiene como objetivo servir de guía detallada para el uso de *EzJavaCode*, una herramienta desarrollada como parte del proyecto de “Desarrollo de Aplicaciones Multiplataforma”. Esta aplicación ha sido diseñada con el propósito de simplificar y automatizar el proceso de creación de clases en el lenguaje de programación Java, mediante una interfaz gráfica interactiva y estructurada que facilita tanto el aprendizaje como la productividad en entornos de desarrollo educativo o profesional.

EzJavaCode permite a los usuarios generar clases Java completas sin necesidad de escribir código manualmente, mediante la definición visual de atributos, métodos y funciones asociadas. Su diseño se basa en principios de usabilidad y eficiencia, permitiendo al usuario centrarse en la lógica de programación y en la organización del código, mientras la herramienta se encarga de la correcta sintaxis y estructura de los elementos definidos. Esta aproximación resulta especialmente beneficiosa para estudiantes de programación orientada a objetos, así como para desarrolladores que buscan acelerar tareas repetitivas o prototipos rápidos.

Entre sus funcionalidades más destacadas se encuentra un módulo para la creación guiada de clases, que permite definir el nombre de la clase, añadir atributos con sus correspondientes modificadores de acceso, características y crear métodos con múltiples parámetros, visibilidad configurable, cuerpo de código personalizado y tipo de retorno. Además, ofrece una funcionalidad de previsualización que permite al usuario inspeccionar el código Java generado antes de su exportación, garantizando así una comprensión completa del resultado.

EzJavaCode también incorpora un sistema de gestión de clases previamente creadas, mediante el cual el usuario puede visualizar, editar o eliminar sus clases desde una única interfaz centralizada. Este sistema contribuye a una mayor organización del trabajo y facilita la reutilización del código generado.

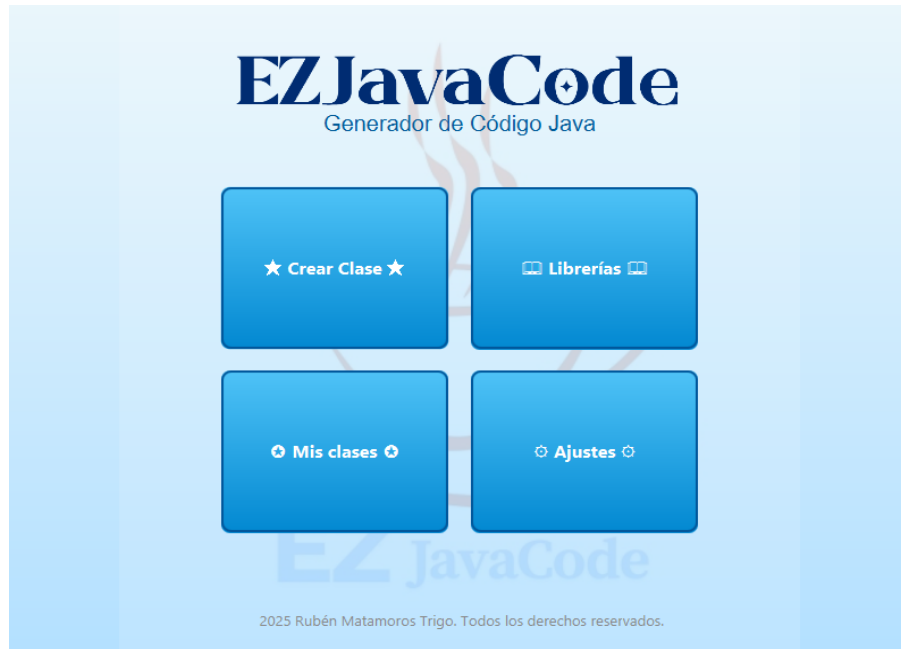
Asimismo, el apartado de ajustes permite personalizar aspectos del entorno de trabajo, como la carpeta de destino por defecto para los archivos exportados, lo que proporciona flexibilidad y adaptación a distintos flujos de trabajo. Finalmente, la herramienta incluye una funcionalidad de librerías personalizadas, donde los usuarios pueden definir funciones reutilizables que podrán ser insertadas rápidamente en distintas clases, promoviendo así el uso de buenas prácticas como la modularidad y la reutilización del código.

En conjunto, *EzJavaCode* constituye una solución práctica y eficaz para apoyar tanto el proceso de enseñanza-aprendizaje de la programación orientada a objetos como el desarrollo de aplicaciones Java a pequeña escala, al ofrecer un entorno visual amigable que reduce la complejidad sintáctica sin sacrificar la estructura formal del lenguaje.

1. CREAR CLASE

Menú Principal.

Pantalla inicial de la aplicación, desde la cual se puede seleccionar una de las funcionalidades principales para acceder a los submenús correspondientes.



Menú Principal - Botón
Crear Clase

Este botón permite acceder a la ventana de creación de clases.



Crear Clase - Nombre de la clase

En esta ventana se debe introducir el nombre de la clase que se desea crear. A continuación, se puede optar por seleccionar '**Crear**' para continuar con el proceso o '**Cancelar**' para descartar. *Todo el ejemplo de creación se desarrollará utilizando la clase 'Persona' como referencia.*



1.1 CREAR CLASE – CREAR ATRIBUTO

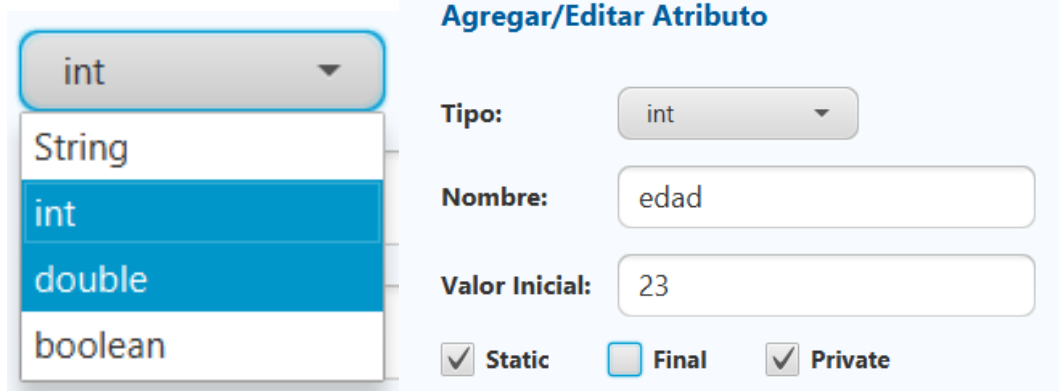
Crear Clase - Crear
Atributos 1

Esta ventana permite crear atributos dentro de la clase seleccionada, en este caso, 'Persona'.



Crear Clase - Crear
Atributos 2

En el apartado '**Agregar/Editar Atributo**' se encuentra un **desplegable** para seleccionar el **tipo de dato** del atributo, un campo de texto para introducir su **nombre** y otro para asignar un **valor inicial** (opcional). Además, es posible configurar la visibilidad y el estado del atributo mediante las opciones '**static**', '**final**' o '**private**'.



Proyecto “Desarrollo de Aplicaciones Multiplataforma”

Título del Proyecto: EzJavaCode



Crear Clase - Crear
Atributos 3

Una vez se hayan introducido todos los valores necesarios para la creación del atributo, se podrá hacer clic en el botón '**Agregar**' para incorporarlo a la tabla.

Crear Atributos de Persona

| Tipo | Nombre | Valor Inicial | Visibilidad |
|------|--------|---------------|----------------|
| int | edad | 23 | private static |
| | | | |
| | | | |
| | | | |

Agregar/Editar Atributo

Tipo:

Nombre:

Valor Inicial:

☐ Static ☒ Final ☐ Private

Crear Clase - Crear
Atributos 4

Para eliminar un atributo de la clase, debe seleccionarse con el ratón y pulsar el botón '**Eliminar**'; el atributo será removido de la tabla de manera definitiva.

Crear Atributos de Persona

| Tipo | Nombre | Valor Inicial | Visibilidad |
|--------|--------|---------------|----------------|
| int | edad | 23 | private static |
| double | altura | 1.85 | final |
| | | | |
| | | | |

Agregar/Editar Atributo

Tipo:

Nombre:

Valor Inicial:

☐ Static ☐ Final ☐ Private

Proyecto “Desarrollo de Aplicaciones Multiplataforma”

Título del Proyecto: EzJavaCode



Crear Clase - Crear
Atributos 5

El botón '**Limpiar**' elimina de forma rápida y sencilla los valores introducidos en los campos. Esta función permite agilizar el proceso, restableciendo los campos a su estado inicial, con el tipo de dato predeterminado establecido en 'String'.

Tipo: double **Nombre:** altura **Valor Inicial:** 1.85
☐ Static ☒ Final ☐ Private **Eliminar** **Limpiar**

Tipo: String **Nombre:** nombreAtributo **Valor Inicial:** (opcional)
☐ Static ☒ Final ☒ Private **Eliminar** **Limpiar**

Crear Clase - Crear
Atributos 6

Una vez se hayan introducido todos los atributos deseados para la clase en Java, se podrá pulsar el botón '**Siguiente**' para acceder a la siguiente pantalla: la creación de métodos o funciones.

Crear Atributos de Persona

| Tipo | Nombre | Valor Inicial | Visibilidad |
|---------|------------|---------------|----------------|
| int | edad | 23 | private static |
| double | altura | 1.85 | final |
| String | genero | masculino | private final |
| boolean | ezjavacode | true | public |

Agregar/Editar Atributo

Tipo: String **Nombre:** nombreAtributo **Valor Inicial:** (opcional)
☐ Static ☐ Final ☒ Private **X Eliminar X** **Limpiar** **✓ Agregar ✓**

← Volver **Siguiente →**

1.2 CREAR CLASE – CREAR METODO

Crear Clase - Crear
Métodos 1

"La siguiente ventana permite crear, editar y eliminar métodos y atributos de la clase Java en desarrollo. En este ejemplo, se trabaja con la clase '*Persona*'.

- **Volver:** permite regresar a la ventana de atributos.



Métodos de Persona

| Nombre | Tipo | Parámetros | Acciones |
|--------------------------|------|------------|----------|
| No hay métodos definidos | | | |

← Volver Añadir Método nuevo Exportar Persona

Crear Clase - Crear
Métodos 2

"El botón '**Añadir método nuevo**' despliega el menú para la creación y desarrollo de métodos o funciones."



Métodos de Persona

+ Añadir método de la librería +

| Nombre | Tipo | Parámetros | Acciones |
|--------------------------|------|------------|----------|
| No hay métodos definidos | | | |

← Volver + Añadir Método nuevo + ✓ Exportar Persona ✓

Crear Clase - Crear
Métodos 3

Esta ventana permite la creación de funciones o métodos. En ella se encuentran todas las opciones necesarias para definir un método o función, que se explicarán a continuación una por una



Crear Metodos/Funciones de Persona

Visibilidad: Elegir tipo:

Nombre de la función:

Parámetros de entrada

| Tipo | Nombre | Eliminar |
|---------------------|--------|----------|
| Tabla sin contenido | | |

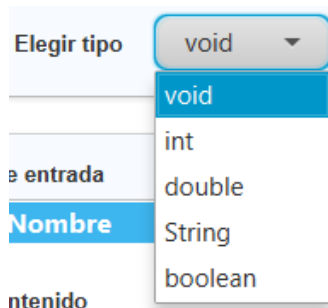
☐ ¿Estático?

Código

Return

Crear Clase - Crear
Métodos 4

En este apartado se encuentran dos desplega-
Visibilidad: permite seleccionar el nivel de acceso del método..
Elegir tipo: indica el tipo de dato que devolverá el método.



Elegir tipo:



Visibilidad:

Crear Clase - Crear
Métodos 5

En este apartado se debe introducir el nombre del método.

Nombre de la función

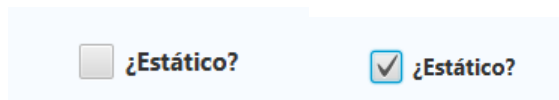
Crear Clase - Crear
Métodos 6

En los campos de parámetros de entrada se deben introducir los datos correspondientes. Tras introducirlos, al pulsar **'Añadir parámetro'**, se registrarán como parámetros de entrada del método. Cada parámetro podrá eliminarse mediante el icono de eliminación (cruz).



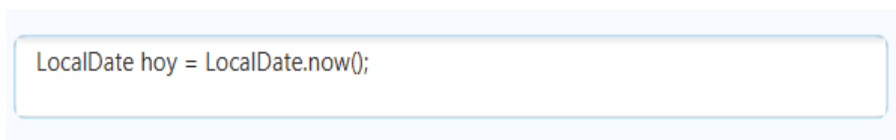
Crear Clase - Crear
Métodos 7

En estos checkboxes se puede seleccionar ajustes del método.



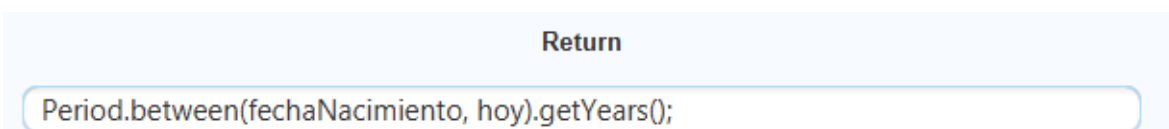
Crear Clase - Crear
Métodos 8

En esta sección se debe incluir el código de la función.



Crear Clase - Crear
Métodos 9

En el caso de tener que devolver algún valor como resultado de la función (En el ejemplo usamos void por lo que no será necesario) deberemos escribir aquí el valor para devolver.



Crear Clase - Crear
Métodos 10

El botón “Incluir Método” permite agregar la función a la tabla de funciones de nuestra clase.

Crear Metodos/Funciones de Persona

Visibilidad: public
Elegir tipo: int

Nombre de la función calcularEdad

Parámetros de entrada

| Tipo | Nombre | Eliminar |
|-----------|-----------------|---|
| LocalDate | fechaNacimiento | X |
| | | |

Tipo
Nombre
Añadir parámetro

☒ ¿Estático?

Código

LocalDate hoy = LocalDate.now();

Return

Period.between(fechaNacimiento, hoy).getYears();

Guardar Cambios

Crear Clase - Crear
Métodos 11

Cuando exista algún método en nuestra clase se desplegarán dos botones, los cuales te permiten editar la clase o eliminarla.
 El botón “Editar” abre un menú similar a “Ventana Crear Método”.
 El botón “Eliminar” permite al usuario restar esa función del listado de Métodos.

| Nombre | Tipo | Parámetros | Acciones | |
|--------------|------|------------|---|---|
| CalcularEdad | void | int edad | Editar | Eliminar |

Crear Clase - Crear
Métodos 12

Al pulsar el boton “Exportar Clase” pasaremos al ultimo paso de la creación de nuestra clase.

← Volver
+ Añadir Método nuevo +
✓ Exportar Persona ✓

1.3 CREAR CLASE – PREVISUALIZACIÓN

Crear Clase - previsualización

Esta ventana permite al usuario ver como es el código que ha generado el programa gracias a los pasos anteriores. El botón “Volver al menú principal”, te permite redirigir al usuario a la primera pantalla del programa.

```
public class Persona {  
  
    private static int edad = 23;  
    private final String altura = "1.85";  
    public final String genero = "masculino";  
    private static final boolean ezjavacode = true;  
  
    public void CalcularEdad(int edad) {  
        int anioActual = Year.now().getValue();  
        int anioNacimiento = anioActual - edad;  
  
        System.out.println("Tienes " + edad + " años, por lo tanto naciste en el año " +  
    }  
}
```

< | >

Volver al menú principal

2. MIS CLASES

Menú Principal - Botón
Mis clases

Este botón permite acceder a la ventana de “Mis clases”.



Mis Clases - Pantalla
Principal

Este menu muestra todas las clases creadas en la carpeta donde el programa este leyendo (Se muestra en la ventana de ajustes). En ella habrá diferentes opciones para poder editar, visualizar o incluso eliminar las clases creadas por el usuario.



Mis Clases - Botón editar

El botón editar accede a un menú similar a crear clase, con la diferencia de que serán cargados todos los datos de la clase seleccionada.

Persona

 **Editar**

 **Visualizar**

 **Eliminar**

Mis Clases - Botón
Visualizar

El botón de visualizar accede a un menu donde se podrá visualizar el código de una manera clara, esta función aporta claridad a los usuarios sobre sus códigos.

Persona

 Editar

 Visualizar

 Eliminar

Código de la clase Persona

```
public class Persona {  
  
    private static int edad = 23;  
    private final String altura = "1.85";  
    public final String genero = "masculino";  
    private static final boolean ezjavacode = true;  
  
    public void CalcularEdad(int edad) {  
        int anioActual = Year.now().getValue();  
        int anioNacimiento = anioActual - edad;  
  
        System.out.println("Tienes " + edad + " años, por lo tanto  
    }  
}
```

Cerrar

Mis Clases - Botón
Eliminar

El botón de eliminar permite al usuario borrar esa clase de la carpeta en su totalidad. Deberá pasar por una pantalla de confirmación antes de ejecutar la acción.

Persona

 Editar

 Visualizar

 Eliminar

Eliminar Persona

¿Estas seguro
de eliminar la clase?

Persona

Cancelar

Aceptar

3. AJUSTES

Menú Principal - Botón
Ajustes

Este botón permite acceder a la ventana de ajustes.



Ajustes - Pantalla
principal

Esta ventana permite al usuario modificar todos los ajustes
respectivos a la aplicación como configuración o
personalización.



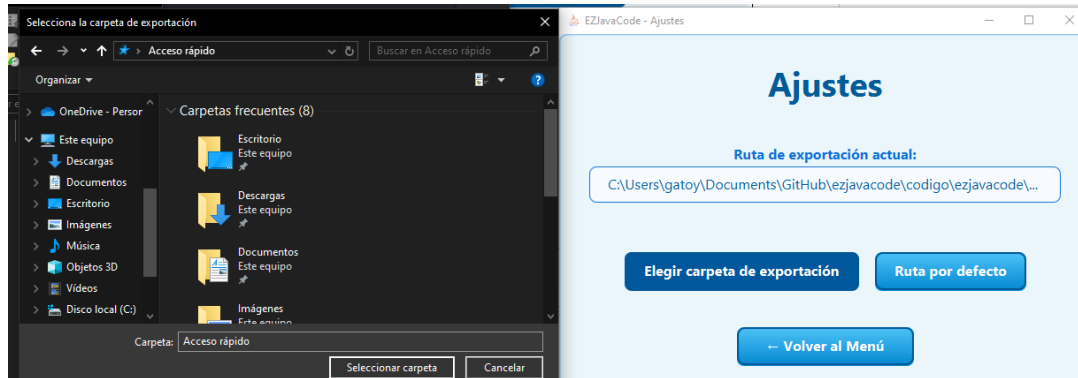
Proyecto “Desarrollo de Aplicaciones Multiplataforma”

Título del Proyecto: EzJavaCode



Ajustes - Botón Elegir carpeta

Permite al usuario seleccionar cual es la carpeta la ubicación en directorio desde donde se exportan y leen los archivos que usará el programa.



Ajustes - Botón Ruta por defecto

El programa por defecto incluye una carpeta donde se exportarán todas las clases creadas. Si el usuario lo desea puede ser modificada la ruta.



4. LIBRERIAS

Menú Principal - Botón
Librerías

Este botón permite acceder a la ventana de “Mis clases”.



Librerías - Mi librería de
Funciones

Esta pantalla permite crear funciones predefinidas por el usuario. Las funciones se podrán crear, editar o eliminar. La intención de estas funciones es que el usuario pueda agregarlas a sus métodos de manera rápida y sencilla en caso de repetición entre clases.

