



Java Development Kit (JDK) y el entorno de desarrollo ECLIPSE

1. Java Development Kit (JDK)

- Descargue la Plataforma JAVA (Edición Estándar o Java SE) desde la URL <http://www.oracle.com/technetwork/java/javase/downloads/index.html> (JDK)

1.1. Creación de una clase Bienvenida que imprima en la consola el mensaje “Hola”

- Abra cualquier editor de texto y defina la clase
- Compile usando el JDK
- Ejecute en la línea de comando
- Pase dos parámetros: nombre y apellido, modifique la clase Bienvenida para que imprima “Hola <apellido> <nombre>”

1.2. Creación de una clase MultiploTest

Escriba una clase MultiploTest la cual desde su método main obtenga 2 parámetros a y b y luego imprima si a es múltiplo de b.

1.3. Creación de una clase Temperatura

Cree una clase Temperatura que tenga un método de clase llamado calcularCelsius, y que dada una temperatura en grados Fahrenheit devuelva la misma en Celsius. Imprima por consola el resultado del método. Para convertir valores Fahrenheit a Celsius, reste 32 al valor Fahrenheit, multiplique por 5 y divida entre 9.

Asegúrese que el resultado obtenido es correcto (por ejemplo realizando los cálculos en forma manual o con una máquina calculadora).

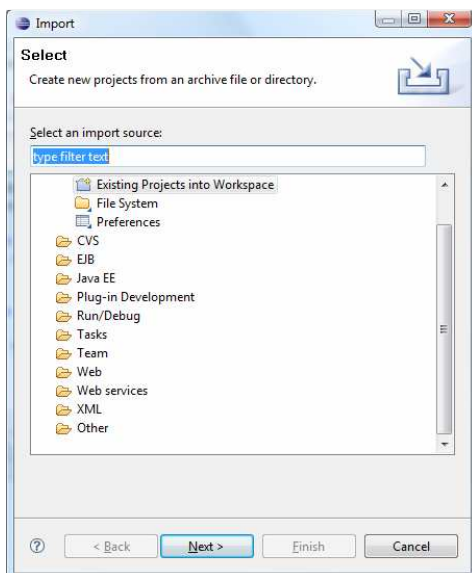
2. El entorno de Desarrollo ECLIPSE

- Descargue Eclipse desde la URL = <http://www.eclipse.org/downloads/>. Seleccione la versión Eclipse Classic y descomprima.
- Descargue el archivo Practica1.zip y descomprima en algún lugar de su disco local.

Eclipse organiza el código fuente en proyectos con el propósito de facilitar el desarrollo de aplicaciones, sin embargo, esto no es un requerimiento del lenguaje Java.

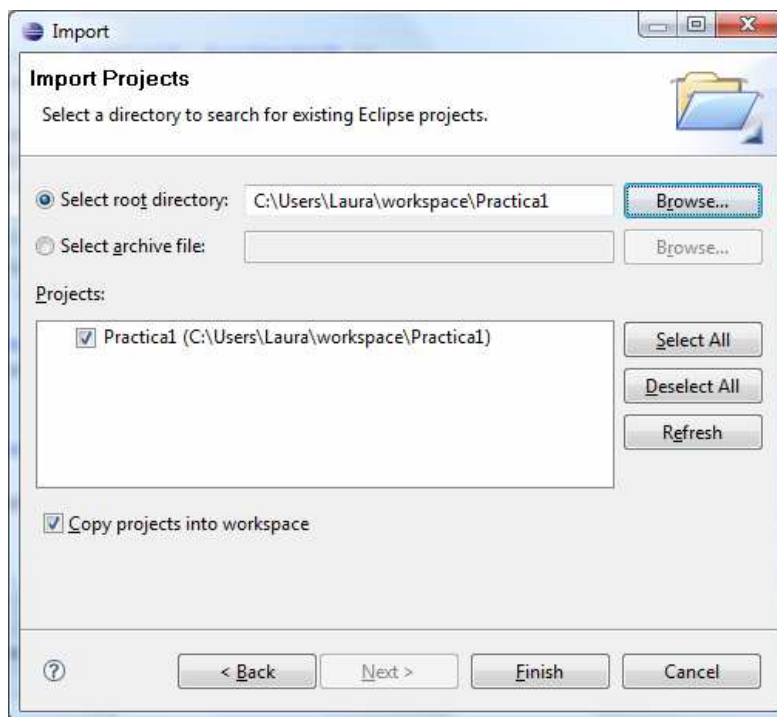
2.1. Creando un Proyecto en Eclipse

- Desde la barra de menú, seleccione File -> Import ...
- Seleccione Existing Projects into Workspace y presione Next.





- Luego seleccione la carpeta Practica1 que descomprimió en su disco. Presione el botón Select All para importar todos los recursos de ese archivo a su proyecto como muestra la figura y presione Finish.

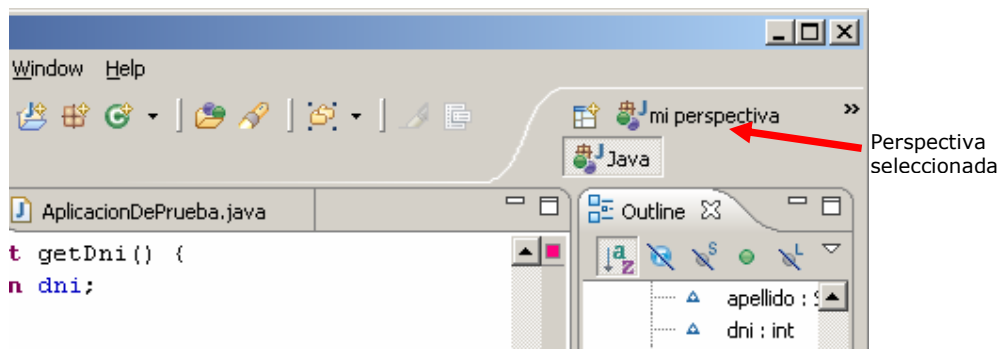


2.2. Usando vistas y perspectivas

Eclipse utiliza el concepto de vistas para permitirnos explorar y modificar los programas, examinar resultados, depurar código, etc. Las perspectivas son organizaciones visuales de una selección de esas vistas. Existen perspectivas preestablecidas y también es posible crear nuevas personalizadas.

Eclipse muestra por defecto la perspectiva Java.

Para cambiar de perspectiva se puede utilizar los botones del margen superior derecho, o desde el menú Window -> Open Perspective



Para cambiar de vista se puede utilizar el menú Window -> Show View

¿Qué otras perspectivas puede abrir?

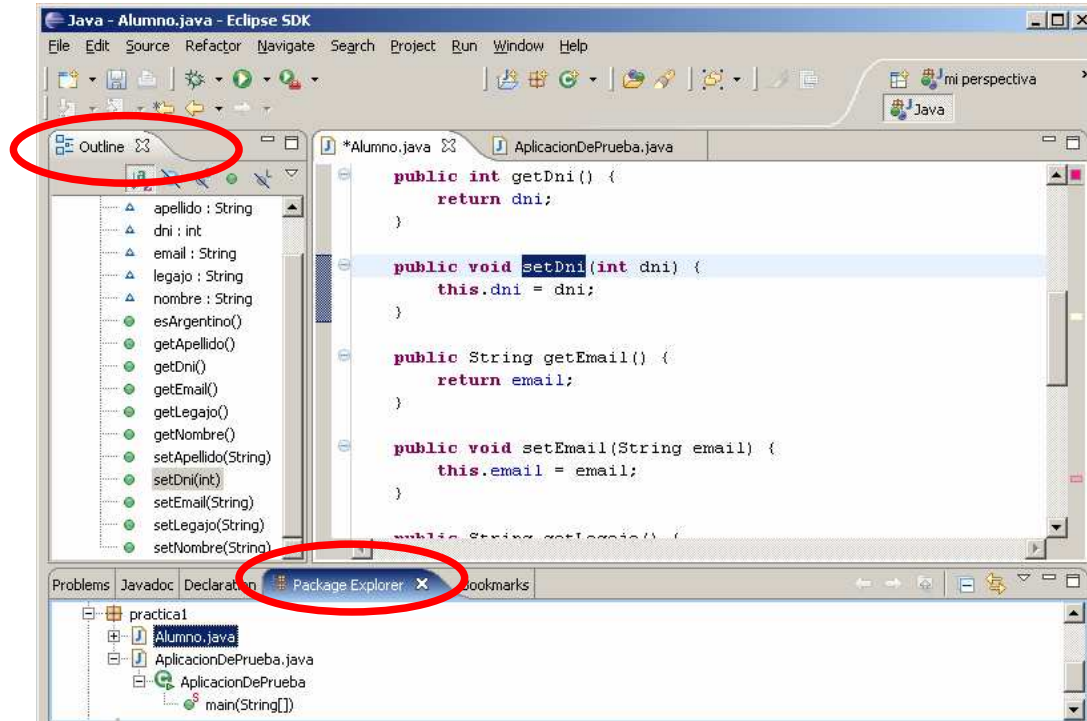
.....

¿Qué vistas puede ver en la perspectiva Java?

.....



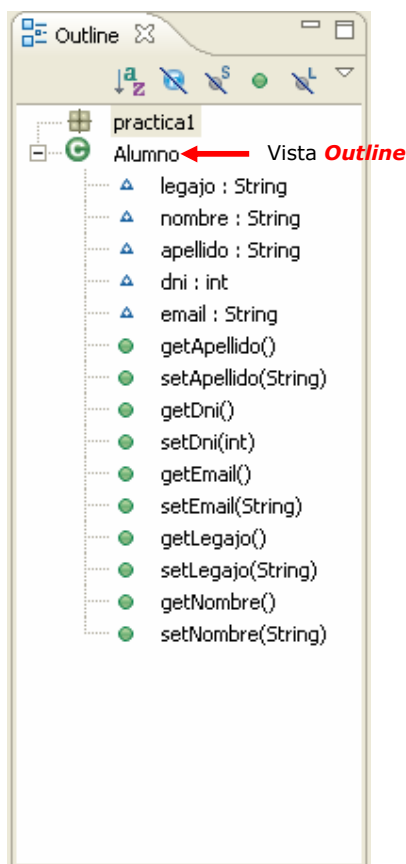
Realice una disposición de vistas como la mostrada abajo



Guarde esta disposición de vistas como una nueva perspectiva llamada “Mi Perspectiva” desde el menú Window -> Save Perspectiva As..

2.3. Explorando paquetes y editando código en el entorno

- En la vista Package Explorer expanda el paquete practical1 para inspeccionar su estructura. Cada paquete contiene uno o más archivos Java.
- Busque la clase Alumno.java y haga doble click para abrirlo en el editor java. Observe el código fuente: estructura de la clase, palabras claves, etc.
- Seleccione la vista Outline. Observe como muestra el paquete al cual pertenece la clase, los paquetes que importa, las declaraciones, miembros, métodos.



El código fuente puede editarse completamente o visualizando sólo un método determinado. Seleccione en el editor la clase Alumno.java y seleccione en la vista Outline un método determinado, haciendo doble clic sobre el mismo se accede directamente a la definición del mismo. Presione el botón Sort, para ver ordenado alfabéticamente los elementos de la vista Outline.

- Cierre el archivo fuente Alumno.java y si hizo cambios no los salve.

2.4. Vista Bookmarks


- En la vista Package Explorer expanda el paquete practica1 y abra el archivo AplicacionDePrueba.java.
- En la barra de menú seleccione Window -> Show View -> Other Se abrirá una ventana con las vistas disponibles. Abra la carpeta Basic y seleccione Bookmarks. Presione OK, y la Vista Bookmark será abierta en la perspectiva Java.
- En el código de la clase AplicacionDePrueba seleccione el encabezado del método: public static void main
- y desde la barra de menú seleccione Edit -> Add bookmark ... y acepte con OK.
- Cierre el archivo AplicacionDePrueba.java. Pruebe hacer doble click sobre las celdas de la Vista Bookmark, para que abra el código fuente y sitúe el cursor en la marca establecida.
- Para borrar el marcador, selecciónelo y presione desde el menú contextual la opción Delete o desde el botón Delete.
- Cierre la Vista Bookmark y el archivo AplicacionDePrueba.java.

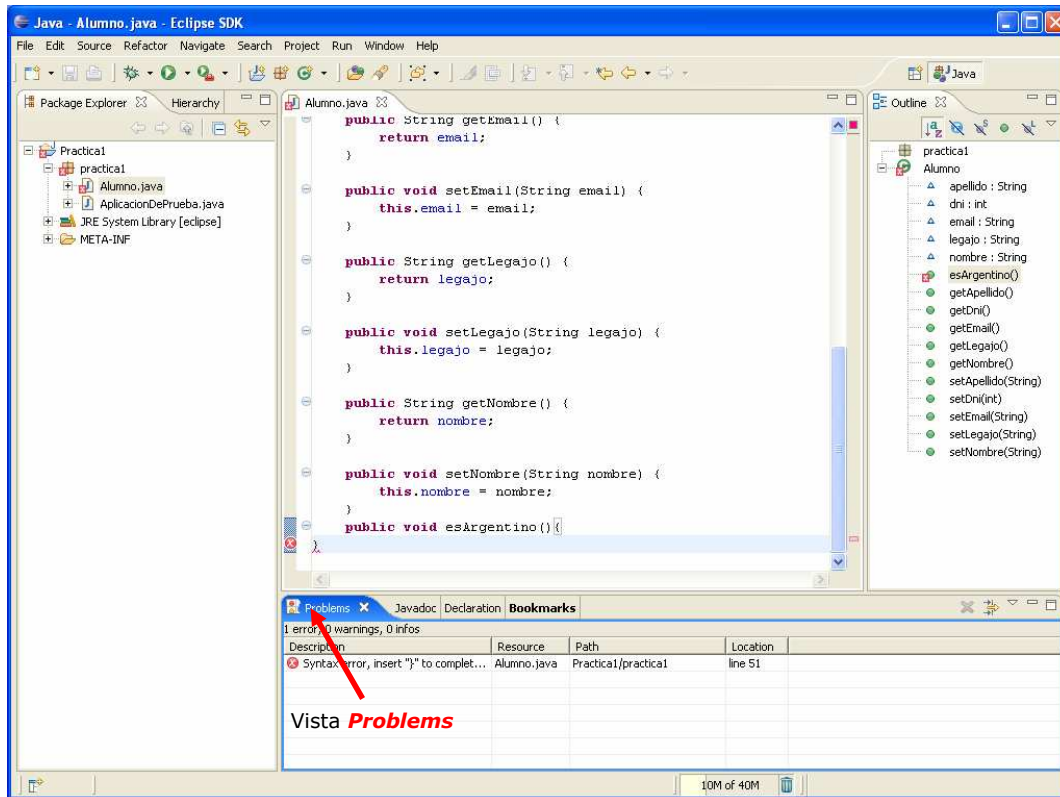
2.5. Usando el asistente de código

- En la perspectiva Java, y desde la vista Package Explorer abra el archivo Alumno.java.
- Asegúrese de que el botón Sort (de la vista Outline), esté presionado para que muestre la vista ordenada alfabéticamente.
- Agregue en el código fuente, al final de la clase el siguiente código:

```
▪ public void esArgentino() {
```
- Observe como automáticamente el código es agregado en orden alfabético



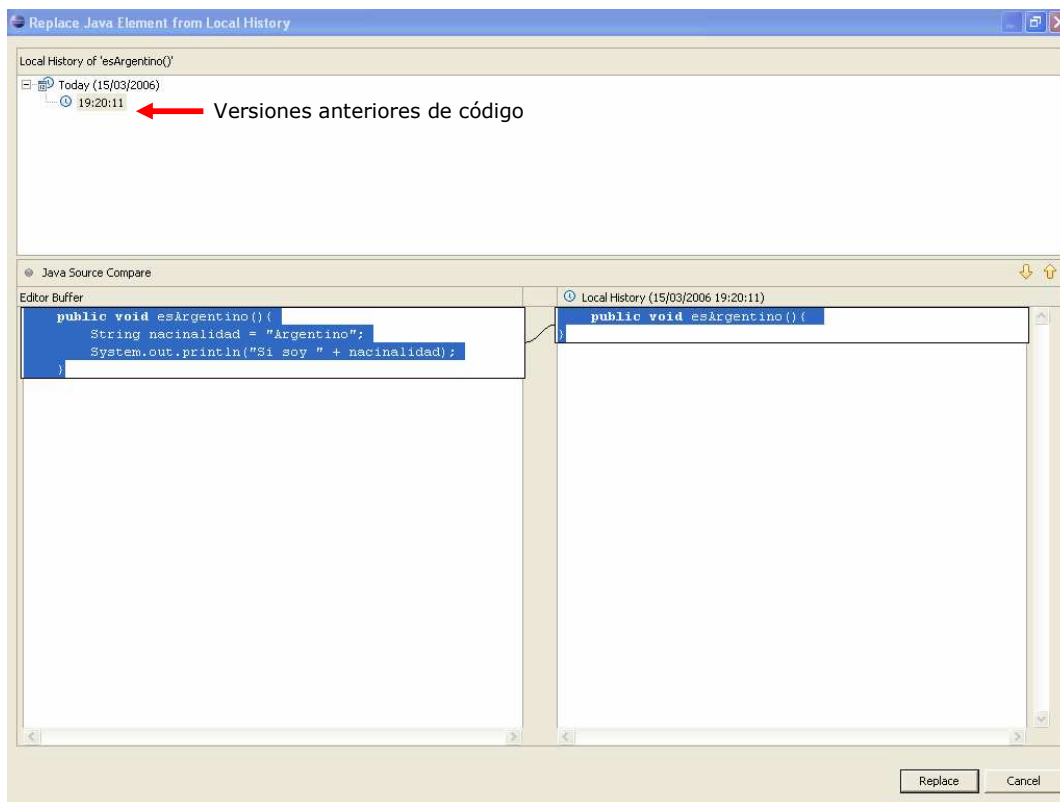
- Presione el botón Save. Eclipse compila automáticamente cuando se salvan los fuentes. Los errores son mostrados en la vista Problems y en el editor con el símbolo: .



2.6. Reemplazo de un método desde el Local History

- Continúe trabajando con la clase Alumno.java. En la vista Outline, seleccione el método esArgentino() que acaba de crear, edítelo y agregue una }. Luego abra el menú contextual y seleccione Delete y confirme.
- Agregue nuevamente un método al final de la clase con el siguiente código:

```
public void esArgentino(){  
    String nacinalidad = "Argentino";  
    System.out.println("Si soy " + nacinalidad);  
}
```
- Salve la clase.
- En la vista Outline, seleccione el método esArgentino() que acaba de crear, abra el menú contextual y seleccione Replace with -> Element from Local History....
- Se abrirá una ventana con las versiones anteriores del método. Cada vez que se elige una versión en el panel superior, los paneles inferiores muestran la versión actual comparándola con la seleccionada en el panel superior.
- Acepte presionando el botón Replace. El código será reemplazado.

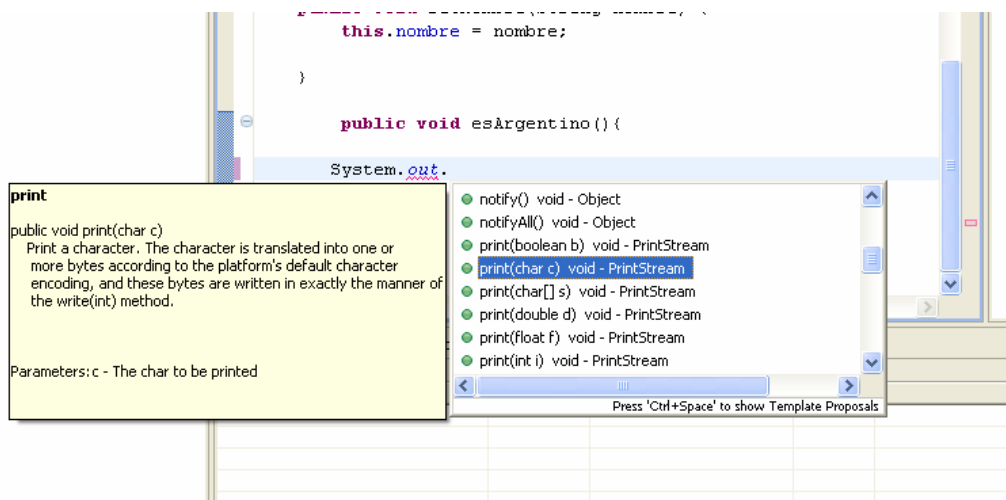


2.7. Usando el asistente de contenido

- Continúe trabajando con la clase Alumno. En la vista Outline, seleccione el método `esArgentino()`. Agregue las siguientes líneas en el método seleccionado:

`System.out.`

- Con el cursor al final de `System.out.` presione `CTRL+Barra espaciadora` para activar el asistente de código.



- Seleccione el método `print(char c)` y presione `ENTER`. Después de que el código es insertado complételo así:

`System.out.print('A');`



- También se puede activar este asistente poniendo la referencia del objeto o una clase y “.”
- Salve la clase.

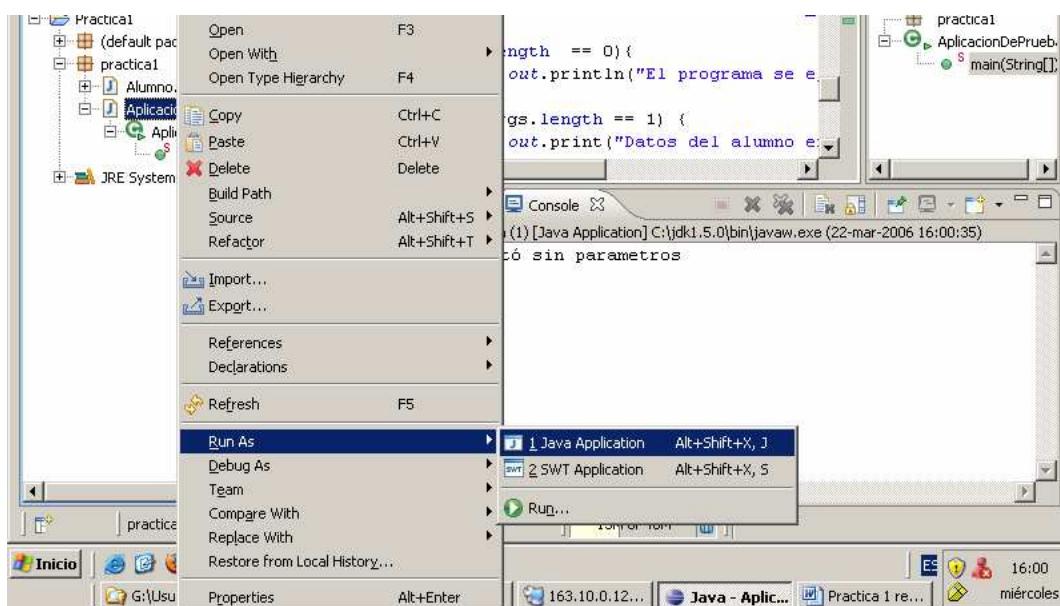
2.8. Corriendo un programa Java

- Para correr un programa Java es necesario tener una clase que contenga un método llamado main cuya declaración sea:

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

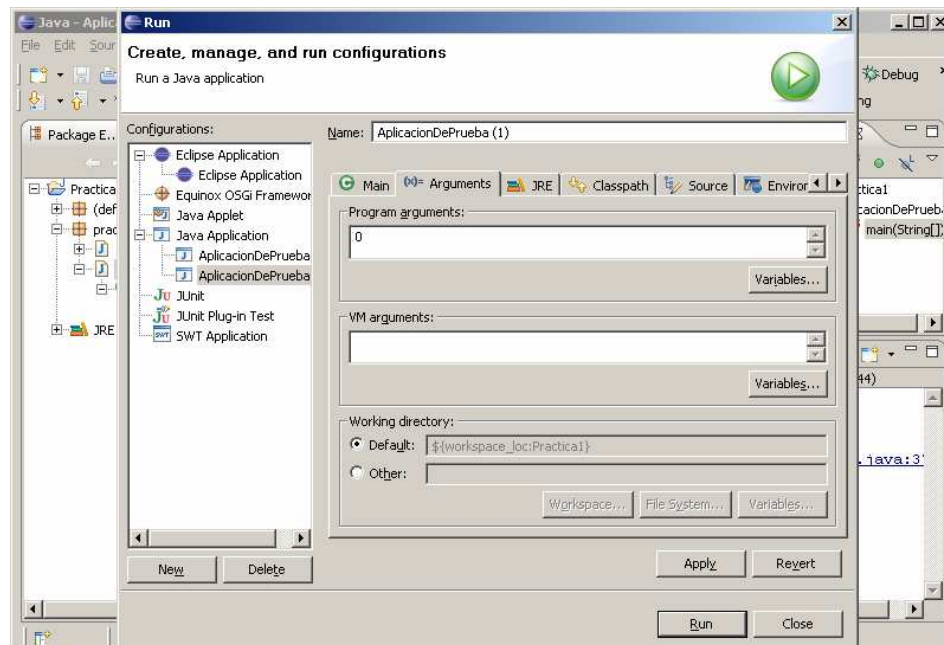
- En esta aplicación ejemplo la clase que posee este método es AplicacionDePrueba.
- Corra la aplicación seleccionando la clase AplicacionDePrueba en la vista Package Explorer y con el botón derecho seleccione Run As -> Java Application

La aplicación crea 2 alumnos e imprime los datos personales del alumno especificado por un numero de orden. Si no se especifica dicho número imprime “El programa se ejecutó sin parámetros”.

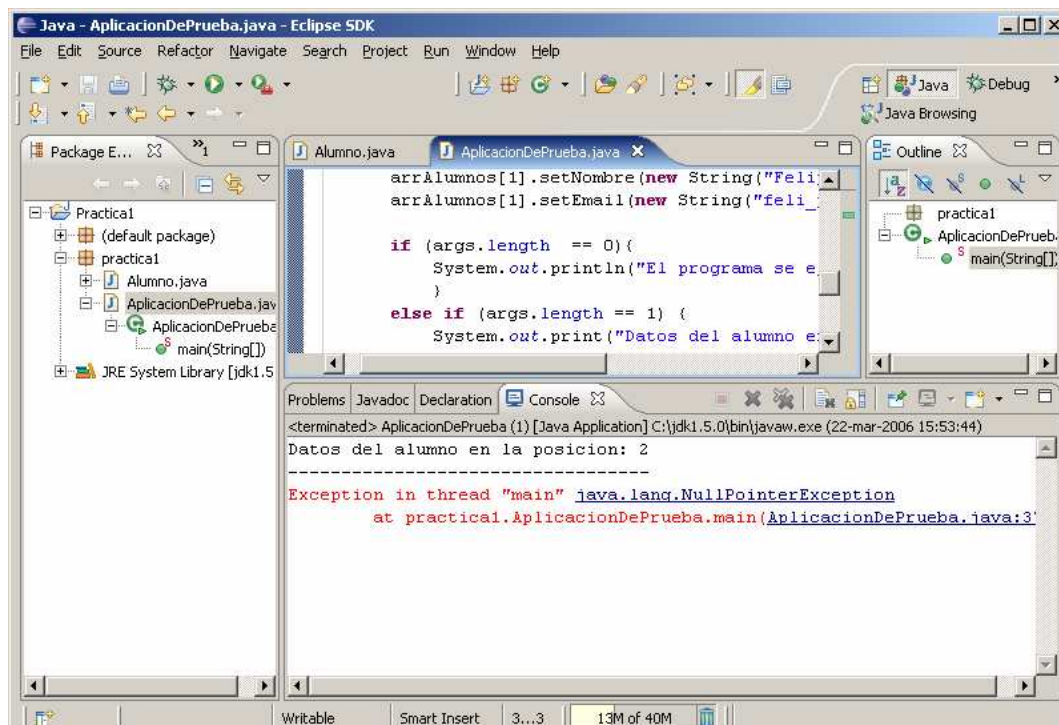


Java utiliza una vista llamada Console para dejar mensajes. Esta aplicación hace uso de ella para obtener una salida de forma sencilla.

- Corra la aplicación pasándole como parámetro el número de orden igual a 0 ó 1 Para esto seleccione el menú Run -> Run... , luego seleccione la pestaña Arguments y en Program Arguments especifique 0 ó 1. El programa debería imprimir los datos personales del alumno seleccionado.



- Pruebe de correr el programa con un número de orden mayor a 2. Esto debería dar un error de ejecución.



2.9. Usando el Debug

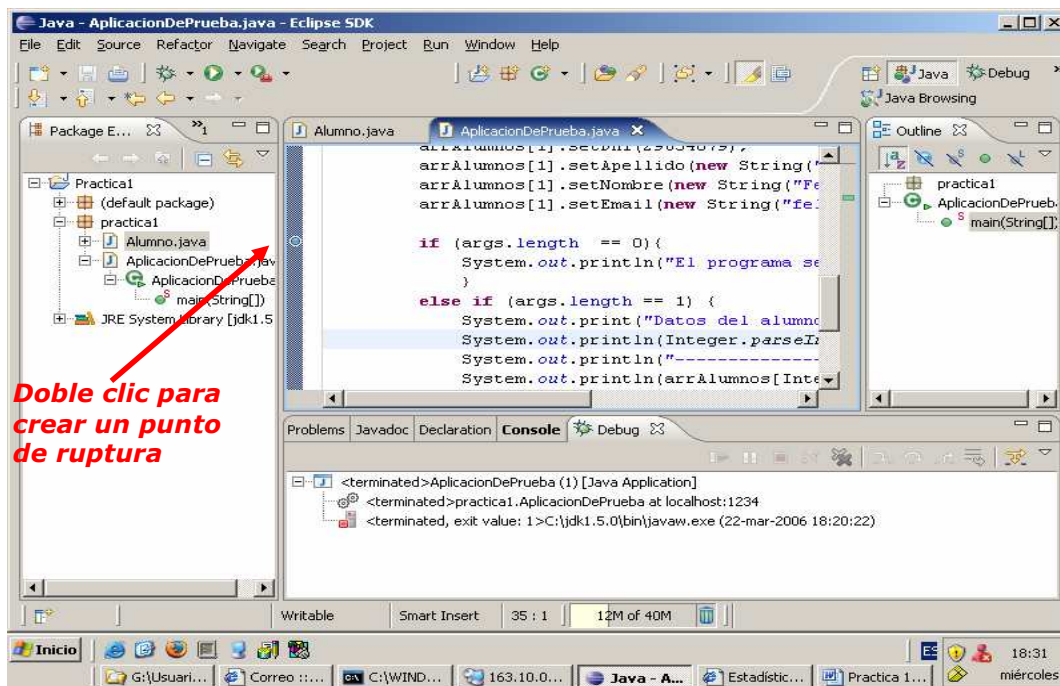
Eclipse provee una perspectiva llamada Debug para encontrar errores que puedan producirse durante la ejecución de un programa. Para esto se pueden establecer puntos de ruptura (Breakpoints) para que a partir de ahí se pueda correr el programa paso a paso y pudiendo examinar el estado de las variables y el flujo de ejecución que sigue el mismo.

- En el código de la clase AplicacionDePrueba coloque un punto de ruptura en la línea que dice:



```
if (args.length == 0){
```

- haciendo doble click en el margen izquierdo sobre esa línea.



- Ejecute el debug seleccionando el menú Run -> Debug... deberá abrir la perspectiva Debug. Examine el contenido de las variables que contienen los datos de los alumnos y ejecute paso por paso el programa.

