

IIP (E.T.S. de Ingeniería Informática)  
Curso 2019-2020

## *Práctica 1. Introducción: Linux, Java y BlueJ*

Profesores de IIP  
Departamento de Sistemas Informáticos y Computación  
Universitat Politècnica de València



### Índice

<b>1. Objetivos y trabajo previo a la sesión de prácticas</b>	<b>2</b>
<b>2. Configuración de los laboratorios del DSIC</b>	<b>2</b>
<b>3. El sistema operativo Linux</b>	<b>2</b>
<b>4. Ficheros y directorios</b>	<b>6</b>
<b>5. Características de los comandos</b>	<b>7</b>
5.1. Formato de los comandos . . . . .	8
5.2. Nombres de ficheros y directorios . . . . .	8
5.3. Uso de comodines y caracteres especiales . . . . .	9
<b>6. Comandos más frecuentes</b>	<b>9</b>
6.1. Obtención de ayuda o información sobre otros comandos . . . . .	10
6.2. Trabajo con directorios . . . . .	10
6.3. Trabajo con ficheros . . . . .	11
<b>7. Edición, compilación y ejecución con BlueJ</b>	<b>13</b>
<b>8. Anexos</b>	<b>17</b>
8.1. Otros comandos útiles en Linux . . . . .	17
8.2. Acceso remoto a los laboratorios . . . . .	18

## 1. Objetivos y trabajo previo a la sesión de prácticas

Con esta primera práctica se pretende que te familiarices con todos los aspectos del entorno de trabajo que utilizarás a lo largo del curso. En concreto, aprenderás a:

- Interactuar con el sistema operativo Linux usando un entorno gráfico.
- Utilizar los comandos básicos del sistema operativo.
- Organizar tu espacio de trabajo en el sistema, estructurándolo mediante directorios.
- Utilizar el entorno integrado de programación BlueJ para editar, compilar y ejecutar un programa Java.

Para la realización de esta práctica no son necesarios conocimientos previos. Si que sería conveniente que leyeras el boletín antes de realizarla en el laboratorio. De esa forma podrás organizar mejor tu tiempo.

## 2. Configuración de los laboratorios del DSIC

A diferencia de otras instalaciones informáticas, mucha de la información que utilices durante tu trabajo (por ejemplo, los programas en Java que puedas escribir) no se almacenan en el ordenador particular con el que trabajes, sino en un sistema virtual propio del DSIC (una nube).

Esto quiere decir que tu visión del sistema (lo que ves cuando trabajas en uno de los puestos) es independiente del ordenador concreto con el que trabajes. Del mismo modo, si te conectas remotamente<sup>1</sup> a los laboratorios del DSIC (por ejemplo, desde tu casa o desde un laboratorio de la Escuela) verás exactamente lo mismo que si trabajas localmente.

Sin embargo, cada usuario del DSIC tiene disponible una carpeta (directorio) especial, denominado **DiscoW** que contendrá la única información de cada usuario que será permanente. Por lo tanto, toda la información que sitúes fuera de dicha carpeta se puede perder, por ejemplo, cuando haya alguna reinstalación del sistema. Es por esto que te recomendamos usar esta carpeta para salvar el material que generes durante las prácticas.

Puedes acceder a la información de la carpeta **DiscoW** en cualquiera de los sistemas instalados en los laboratorios del DSIC: Linux y Windows<sup>2</sup>, por lo que podrás acceder a tu información, sea cual sea el sistema que utilices. Recuerda que dicha información no aparecerá en tu usuario si te conectas en otras ubicaciones sin acceder remotamente al DSIC, por ejemplo, desde los laboratorios de la ETSInf.

## 3. El sistema operativo Linux

Un *Sistema Operativo* (SO) es un conjunto de programas que permiten una gestión eficiente de los elementos del ordenador, simplificando la realización de las operaciones

---

<sup>1</sup>En el Anexo 8.2 se explica como hacerlo

<sup>2</sup>En el sistema Windows instalado en los laboratorios del DSIC, tu carpeta personal **DiscoW** se encuentra en una unidad llamada **homes(\\fileserver.dsic.upv.es)(W:)**

más relevantes de los mismos. Permite, por ejemplo, manipular de forma sencilla los dispositivos de almacenamiento; los dispositivos de interacción con los seres humanos, como ratones, teclados, pantallas, etc; elementos de comunicaciones y conexión con otros ordenadores; y un largo etc.

Algunas de las funciones más importantes de los sistemas operativos son:

- Gestionar los distintos recursos del ordenador: almacenamiento, entrada y salida, procesador central, comunicaciones, etc.
- Gestionar accesos simultáneos de varios usuarios que, gracias a ello, tienen la impresión de trabajar con un ordenador individual, dedicado a cada uno de ellos.
- Controlar un acceso seguro al sistema, de forma que los usuarios puedan acceder y manipular en el sistema sólo aquello para lo que están autorizados.
- Facilitar la interacción con la máquina, proporcionando herramientas gráficas y textuales que permitan la manipulación de la misma de la forma deseada.

El SO que utilizarás generalmente en el laboratorio de prácticas es el Linux; es un SO gratuito (sus fuentes pueden conseguirse, y alterarse, por ejemplo) y potente (debido a que es una versión de Unix), que se encuentra muy extendido tanto en ámbitos académicos, empresariales y oficiales; de hecho, hay un plan de la administración española para utilizarlo de forma intensiva. De las múltiples distribuciones disponibles de Linux, en los laboratorios hay instalado este curso Linux CentOS 7.5.

Al igual que en otros SO (como Windows o Mac OS X), la interacción con Linux puedes realizarla de dos maneras:

- En modo textual o *consola*: escribiendo mediante el teclado órdenes al sistema, que se denominan *comandos* y que son ejecutados por el mismo.
- En modo gráfico: usando entornos gráficos (por ejemplo MATE, GNOME, KDE, etc.) que permiten utilizar el ratón y el teclado para interactuar a través de ventanas, menús, aplicaciones, etc. En estas prácticas utilizaremos el entorno MATE que es el instalado por defecto en los laboratorios.

## Actividad #1

1. Enciende el ordenador y selecciona la opción para utilizar el SO Windows (esto es necesario para validar tu contraseña, aunque luego pasarás a Linux).
2. Tras una serie de mensajes de arranque, aparecerá una pantalla en la que debes introducir tu nombre de usuario (*login*) y tu clave de acceso (contraseña o *password*).
  - Usuario: la palabra que precede al símbolo @ en tu email, p.e., si tu email fuera **alumniip@inf.upv.es**, tu nombre de usuario sería **alumniip**.
  - Clave<sup>3</sup>: los dígitos de tu DNI (sin letra<sup>4</sup>).

La primera vez que inicies la sesión, el sistema te pedirá que cambies la contraseña. Después de cambiarla, el sistema cierra la sesión.

---

<sup>3</sup>Si ya estabas matriculado en el curso anterior, mantienes tu anterior contraseña

<sup>4</sup>Los alumnos extranjeros tienen un código identificativo que sí lleva letra, del estilo X5567322

3. Sal de Windows y elige la opción del SO Linux (Linux Estándar); cuando acabe el proceso, el sistema te pedirá tu usuario y contraseña; debes usar la nueva contraseña.
4. Tras introducir tus datos, la pantalla presentará un aspecto similar al de la Figura 1, un escritorio en el que puedes observar: una barra con menús desplegables y el icono del navegador Firefox (entre otros), en la parte superior; los iconos **Equipo**, **Carpeta personal de ...**, **thinclient\_drives**<sup>5</sup> y **Papelera**, a la izquierda de la parte central; una barra con los iconos de escritorio, en la parte inferior. Haz clic sobre el menú **Aplicaciones** y navega por los submenús, observando las distintas aplicaciones instaladas.

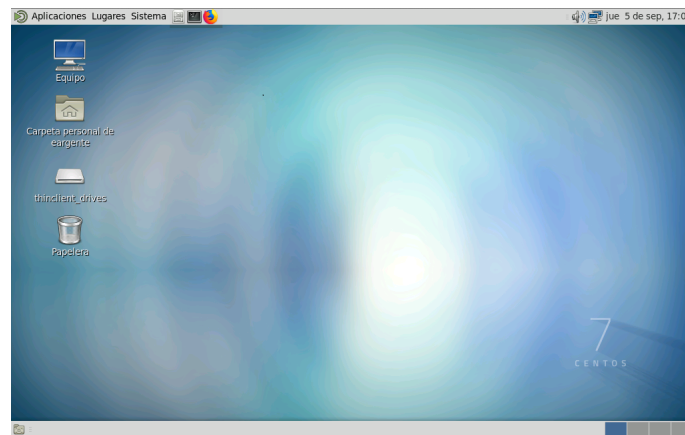


Figura 1: Escritorio de MATE.

5. Haz doble clic sobre el icono **Carpeta personal de ...** del escritorio. Observa que se abre una ventana con los contenidos de la carpeta de usuario, como en la Figura 2(a). Mediante los iconos, el menú o el botón derecho del ratón se puede gestionar dicho contenido (copiar, pegar, mover, eliminar, etc.).

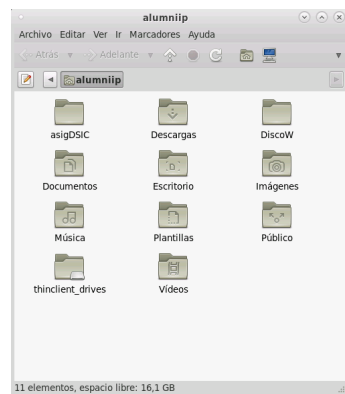
Entre las carpetas de la carpeta de usuario se encuentran **DiscoW** y **asigDSIC**<sup>6</sup>.

6. Accede a PoliformaT desde el navegador Firefox y descarga en tu **DiscoW** el fichero **miFichero.txt**<sup>7</sup> disponible en la carpeta **Recursos/Laboratorio/Práctica 1**. Comprueba que el fichero se encuentra ahora en tu **DiscoW**, como en la Figura 2(b).

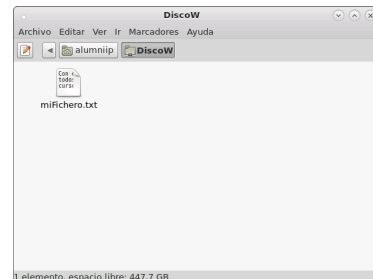
<sup>5</sup> Acceso a los dispositivos de la máquina local como, por ejemplo, las unidades usb

<sup>6</sup> Grupo de carpetas compartidas de diversas asignaturas del DSIC

<sup>7</sup> Selecciona el fichero y pulsa el botón derecho del ratón. Selecciona la opción *Guardar enlace como...* del menú contextual



(a) Carpeta de usuario.



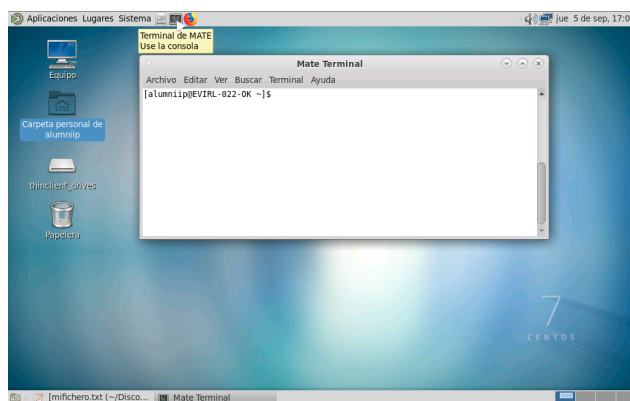
(b) Carpeta DiscoW.

Figura 2: Carpeta de usuario y, dentro de ella, DiscoW.

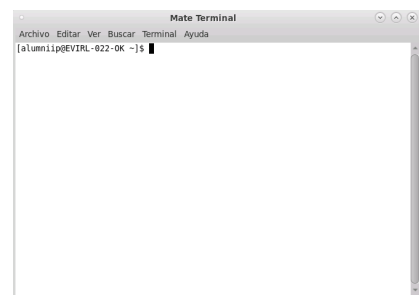
Para interactuar con el SO a través de la línea de comandos, se necesita una aplicación (*terminal* o *consola*) que ejecuta un programa llamado *shell* o *intérprete de comandos*.

## Actividad #2

1. Inicia un terminal seleccionando el icono correspondiente de la barra superior (Figura 3(a)) y te aparecerá una ventana, semejante a la de la Figura 3(b), con un indicador (*prompt*) que consiste en tu nombre de usuario, el identificador de la conexión mediante la que trabajas, el directorio actual y el carácter \$. Después del \$ escribiremos las órdenes a ejecutar por el sistema.



(a) Localización del terminal.



(b) Terminal.

Figura 3: Inicio de un terminal de MATE.

En general, un comando tiene la forma: `comando -opciones argumentos`. Después de escribir el nombre del comando (y posibles opciones y argumentos), éste es ejecutado pulsando la tecla *Enter*. Por supuesto, para ello tenemos que conocer qué comandos existen y para qué sirven. Pero antes veamos cómo se organiza la información en el sistema.

## 4. Ficheros y directorios

El elemento básico de información en el sistema es el *fichero*. A grandes rasgos, un fichero es una secuencia de información homogénea que reside en el ordenador; por ejemplo, es posible encontrar ficheros de caracteres, de valores numéricos, de instrucciones ejecutables por el procesador y, en general, de cualquier tipo de información codificable.

Desde el punto de vista del SO, un fichero tiene un nombre, que permite reconocerlo, y un grupo de atributos, tales como su tamaño, la fecha de su última modificación, su propietario, así como ciertos *privilegios de acceso*, esto es, una definición de lo que pueden hacer con él diferentes usuarios.

El SO también reconoce otro tipo de elemento que permite, a su vez, agrupar otros elementos de información; se denominan *directorios* (o *carpetas*) y tienen precisamente la funcionalidad que parece indicar su nombre, esto es, contener ficheros y/u otros directorios.

Existe un directorio, denominado *raíz*, que engloba jerárquicamente al resto de directorios y ficheros del sistema y se representa mediante el símbolo / (símbolo de la división).

En realidad, el conjunto de directorios y ficheros del sistema conforman una estructura jerárquica en la que unos directorios pueden contener tanto ficheros como, tal vez, otros directorios que a su vez pueden tener una funcionalidad similar (esto es, contener nuevos directorios y ficheros). Esta estructura jerárquica recibe el nombre de *árbol de directorios del sistema* y se puede representar como un árbol invertido, con raíz el directorio /, y cuyos elementos son directorios y/o ficheros; en la Figura 4 podemos ver una parte del árbol de directorios de nuestro sistema (nota que hay un **DiscoW** para cada alumno).

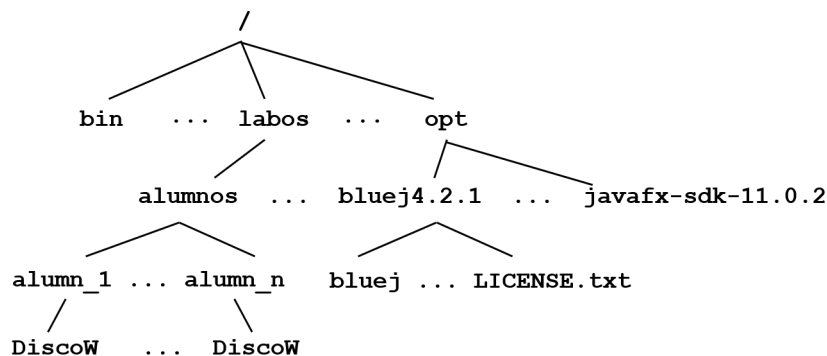


Figura 4: Árbol de directorios.

Cada fichero o directorio del sistema queda representado de forma única estableciendo para el mismo la sucesión de directorios que es necesario recorrer en el árbol de directorios para llegar hasta él a partir del directorio raíz /. Este camino recorrido para llegar al elemento correspondiente, recibe generalmente el nombre de *ruta* (en inglés *path*). Por ejemplo, la secuencia /opt/bluej4.2.1/LICENSE.txt indica todo el recorrido (*path*) necesario para llegar al fichero LICENSE.txt.

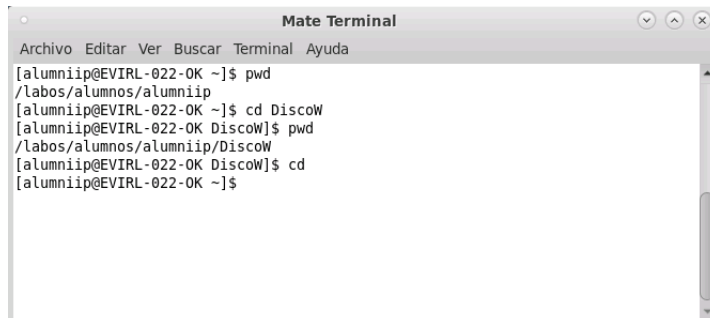
Cualquier usuario del sistema se encuentra, en un momento determinado, situado en uno de los directorios del sistema, este directorio recibe el nombre de *directorio de trabajo* (*working directory*) o *directorio actual*. Las operaciones que el usuario realice relativas a la manipulación de ficheros y directorios se efectuarán, por defecto, sobre los elementos de dicho directorio de trabajo. El usuario puede, además, cambiar su directorio de trabajo (usando el comando `cd`).

Al iniciarse una sesión de trabajo, automáticamente el sistema te sitúa en un directorio predeterminado, el *home*. La ruta de este directorio se guarda en una variable del entorno de cada usuario (`$HOME`).

Así, por ejemplo, si un usuario `alumniip` ejecuta el comando `pwd` (acrónimo de *print working directory*) después de haber iniciado la sesión, el sistema muestra `/labos/alumnos/alumniip`, el directorio en el que se encuentra en ese momento el usuario, es decir, su *home*.

### Actividad #3

1. Comprueba en qué directorio te encuentras ejecutando el comando `pwd`.
2. Cambia con el comando `cd DiscoW` al directorio `DiscoW`. Compruébalo con `pwd`.
3. Con el comando `cd` vuelve de nuevo a tu `$HOME`. Observa que el *prompt* cambia, indicando en qué directorio te encuentras en cada momento (Figura 5).



```
Mate Terminal
Archivo Editar Ver Buscar Terminal Ayuda
[alumniip@EVIRL-022-OK ~]$ pwd
/labos/alumnos/alumniip
[alumniip@EVIRL-022-OK ~]$ cd DiscoW
[alumniip@EVIRL-022-OK DiscoW]$ pwd
/labos/alumnos/alumniip/DiscoW
[alumniip@EVIRL-022-OK DiscoW]$ cd
[alumniip@EVIRL-022-OK ~]$
```

Figura 5: Cambio del *prompt* del sistema.

## 5. Características de los comandos

Antes de describir algunos de los comandos más habituales, conviene conocer algunos aspectos importantes relativos al uso de los mismos: el primero consiste en el formato de los comandos, el segundo trata de la forma en que se nombran los ficheros y directorios y, por último, examinaremos cómo poder hacer referencia a un conjunto de ficheros simultáneamente.

## 5.1. Formato de los comandos

Los comandos son palabras individuales que el sistema reconoce y ejecuta y que pueden tener argumentos modificadores (opciones) que alteran de alguna forma el comportamiento del mismo.

### Ejemplos:

- La orden `ls` lista los nombres de los ficheros del usuario que se encuentran en el directorio de trabajo.
- La orden `ls miFichero.txt` lista el nombre del fichero del usuario `miFichero.txt` si se encuentra en el directorio de trabajo.
- La orden `ls -l -a` lista toda la información (modificador `-l`) asociada a los nombres de los ficheros del usuario y del sistema (modificador `-a`) que se encuentran en el directorio de trabajo. Esta orden también se puede escribir como `ls -la`

En Linux es posible escribir parcialmente los comandos, de forma que el intérprete de comandos intente acabarlos automáticamente. Para ello, después de escribirse el comienzo de un comando, debe pulsarse la tecla de tabulación. Con ello, el intérprete intentará acabar el comando (lo que ocurrirá si en el mismo no hay ambigüedades).

**Ejemplo:** la orden `ls -l mi` está incompleta si se desea conocer todas las características del fichero `miFichero.txt`. Si se escribe `ls -l mi` y, a continuación, se teclaea el tabulador, entonces el intérprete de comandos intentará acabar el comando anterior, de forma que si el único fichero que empieza por `mi` es el deseado, entonces el comando aparecerá escrito como `ls -l miFichero.txt`.

Para facilitar la escritura de órdenes se pueden usar las flechas arriba y abajo del teclado (o las combinaciones `Ctrl+P` y `Ctrl+N`) que permiten acceder al historial de comandos utilizados.

### Actividad #4

1. Ejecuta las órdenes `ls` de los ejemplos anteriores desde tu **DiscoW** y comprueba que el resultado es el esperado.
2. Pulsa las flechas arriba y abajo del teclado y observa como tras el *prompt* aparecen los comandos que has introducido anteriormente.

## 5.2. Nombres de ficheros y directorios

En Linux se puede nombrar un fichero o un directorio en un comando de dos maneras:

- **Absoluta:** consiste en anteponer el nombre del mismo con la ruta para llegar a este desde el directorio raíz. Por ejemplo, la orden `ls /opt/bluej4.2.1` lista el nombre de todos los ficheros existentes en el directorio que se encuentra en el lugar indicado (`/opt/bluej4.2.1`).



- **Relativa:** consiste en indicar cómo llegar desde el directorio en el que se está trabajando –determinando todos los directorios por los que es necesario pasar a través del árbol de directorios–, al recurso (fichero o directorio) deseado. Por ejemplo, si el directorio actual es `/opt` y se ejecuta la orden `ls -l bluej4.2.1/LICENSE.txt` se pide al sistema que liste toda la información (modificador `-l`) asociada al nombre de fichero `LICENSE.txt`, al que se debe llegar a través de los directorios indicados.

También se pueden utilizar las siguientes referencias especiales:

- `..` para referenciar al directorio antecesor (padre) de otro dado.
- `.` para referenciar al directorio actual.

## Actividad #5

1. Utiliza el comando `ls` para listar toda la información asociada al contenido del directorio `/opt` usando la referencia `..` para nombrar `/opt` desde tu `$HOME`.

## 5.3. Uso de comodines y caracteres especiales

Se denominan *comodines* a algunos caracteres especiales que se usan en la construcción de los nombres de ficheros y directorios de los comandos y que tienen un significado especial. Su uso permite nombrar conjuntos de ficheros que tienen una característica sintáctica común, permitiendo la aplicación de un mismo comando a un grupo de ficheros o directorios, en lugar de a uno individual.

Los comodines más relevantes son:

- `?` que representa cualquier carácter; por ejemplo, la orden `ls miFicher?.txt` listará el nombre de todos los ficheros con un único carácter (sea éste el que sea) en el lugar del `?`.
- `*` que representa cualquier secuencia de caracteres, incluso la vacía; por ejemplo, la orden `ls miFichero*` listará el nombre de todos los ficheros cuyo nombre empiece por la secuencia `miFichero`.

Si en alguna ocasión se desea que el sistema interprete un carácter como tal (y no de forma especial, como por ejemplo ocurriría con un comodín) es necesario escribirlo entre comillas dobles; por ejemplo, la orden `ls pract2.*` listará el nombre, si existe, de un fichero cuyo nombre en el sistema es exactamente `pract2.*`

## 6. Comandos más frecuentes

A continuación se muestran algunos de los comandos más relevantes agrupados por su funcionalidad, describiéndose cada uno de ellos, de forma breve.

## 6.1. Obtención de ayuda o información sobre otros comandos

Los comandos que siguen acceden a la base de datos de información de comandos del sistema para obtener ayuda o información sobre otros comandos.

- `whatis comando` muestra, de forma breve, para qué sirve el `comando`.
- `apropos comando` muestra las entradas de la base de datos donde aparece `comando`.
- `man comando` muestra la ayuda disponible del `comando` en el manual del sistema. Para salir de la ayuda basta con pulsar la tecla `q`.

## Actividad #6

1. Consulta la información del comando `ls` utilizando `whatis`, `apropos` y `man`.

## 6.2. Trabajo con directorios

Es importante mantener bien estructurada tu información; esto lo comprobarás cuando tengas que gestionar cientos o, tal vez, miles de ficheros. Por lo tanto, en el sistema que utilizas en prácticas, debes mantener bien organizada la información de tu `DiscoW`. Acuérdate, sin embargo, que el resto de información en tu `$HOME` (incluyendo lo que dejes en el Escritorio) no tiene por qué persistir.

Los siguientes comandos te ayudarán a estructurar tu espacio de trabajo:

- `pwd` muestra el camino absoluto desde la raíz al directorio actual.
- `ls directorio` lista los ficheros y directorios de `directorio`, por defecto el de trabajo. Con la opción `-a` muestra los ficheros ocultos (del sistema –aquellos cuyo nombre empieza por `.` –); con la opción `-l` muestra más información sobre cada ítem (sus atributos). Para cada fichero del directorio de trabajo se indica el tipo de fichero (`d` significa directorio y `-` fichero normal); los permisos de acceso (lectura `r`, escritura `w` y ejecución `x`) para el usuario propietario, su grupo y el resto de usuarios, respectivamente; número de vínculos; propietario; grupo; tamaño en bytes; fecha y hora de su última actualización; y su nombre.
- `mkdir directorio` crea un directorio vacío de nombre `directorio`.
- `du directorio` muestra la ocupación de disco de cada fichero y subdirectorio del directorio especificado. Con la opción `-c` muestra un total; con la opción `-h` añade una letra indicativa del tamaño (p.e., M de Megabytes); con la opción `--exclude=nomDir` no incluye la ocupación del directorio `nomDir`.
- `cd directorio` cambia el directorio de trabajo al directorio especificado. Por defecto (si no se indica `directorio`) cambia al `$HOME`. Si `directorio` es `..` cambia el directorio actual al directorio padre.
- `rmdir directorio` borra el `directorio` especificado si está vacío.
- `tree directorio` muestra el contenido del `directorio` en forma de árbol.

## Actividad #7

1. Crea en tu DiscoW el directorio **pruebasIIP**. Dentro de este último, crea el subdirectorio **pract1**.
2. Sitúate en **pract1** y, con una sola orden, crea el subdirectorio **pract1Bis** en **pruebasIIP**.
3. Sitúate en tu DiscoW y, con una sola orden, crea el subdirectorio **prueba** en **pruebasIIP**.
4. Lista el contenido del directorio **pruebasIIP**, incluyendo ficheros ocultos. El resultado será semejante al que sigue:

```
total 20
drwxr-xr-x  5 alumnipp alumnos 4096 sep  4 15:47 .
drwx----- 33 alumnipp alumnos 4096 sep  4 15:45 ..
drwxr-xr-x  2 alumnipp alumnos 4096 sep  4 15:45 pract1
drwxr-xr-x  2 alumnipp alumnos 4096 sep  4 15:47 pract1Bis
drwxr-xr-x  2 alumnipp alumnos 4096 sep  4 15:47 prueba
```

5. Muestra el total en Megabytes que ocupa tu directorio **DiscoW**.

### 6.3. Trabajo con ficheros

Los comandos que siguen sirven para manipular ficheros.

- **cp fichOrigen fichDestino** copia el fichero **fichOrigen** al fichero **fichDestino**. Si **fichDestino** es un directorio existente, se copia el fichero **fichOrigen** a dicha carpeta, manteniendo el nombre original.
- **mv fichOrigen fichDestino** cambia el nombre del fichero o directorio **fichOrigen** que pasará a llamarse **fichDestino**. Si **fichDestino** es un directorio, mueve el fichero **fichOrigen** a dicho directorio.
- **rm fichero** borra el fichero indicado, con la opción **-i** se pide confirmación al usuario. No borra directorios, pero con la opción **-r** borra el directorio especificado y todo su contenido (tanto de ficheros como de directorios).
- **cat fichero(s)** concatena los ficheros que recibe como argumentos y visualiza su contenido en la salida estándar. Si algún fichero no existe, devuelve un mensaje de error.
- **more fichero(s)** visualiza el contenido del fichero por páginas (lo cual no es posible con **cat**). Cada vez que pulsamos el espaciador avanzamos una página, mientras que para retroceder podemos pulsar la tecla **b**. Podemos salir en cualquier momento con **q**.

- **less fichero(s)** es similar al comando **more**, aunque más potente, permitiendo visualizar el contenido del fichero página a página, pero con desplazamientos hacia adelante y hacia atrás; búsqueda de patrones en el texto, etc. Este comando es exclusivo del Linux.
- **wc fichero(s)** muestra el número de líneas, palabras y caracteres del fichero especificado. Si se especifica más de un fichero, despliega los resultados de cada fichero individual y el total de todos los ficheros. Para obtener resultados individuales (líneas, palabras o caracteres) se puede utilizar la opción **-l**, **-w** o **-c**, respectivamente.

## Actividad #8

1. Situado en tu **DiscoW**, comprueba que el árbol de directorios del mismo es el que se muestra en la Figura 6.

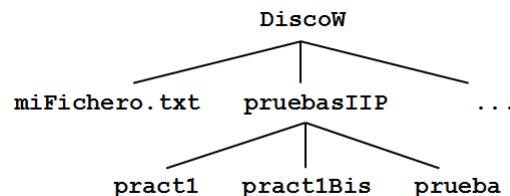


Figura 6: Árbol de directorios de tu **DiscoW** antes de la actividad #8.

2. Copia el fichero **miFichero.txt** al directorio **pruebasIIP/pract1**. Comprueba el contenido de dicho directorio.
3. Borra el fichero **miFichero.txt** de tu **DiscoW**.
4. Sitúate en **pruebasIIP/pract1** y crea una copia del fichero **miFichero.txt** llamada **miFichero2.txt**. Comprueba el contenido del directorio y, usando el comando **cat**, visualiza el contenido de los ficheros **miFichero.txt** y **miFichero2.txt**.
5. Con una sola orden, mueve el fichero **miFichero2.txt** al directorio **pruebasIIP/prueba**. Comprueba el contenido de los subdirectorios **pract1** y **prueba**.
6. Sitúate en **pruebasIIP/prueba**; cambia el nombre del fichero **miFichero2.txt** por **miFich.txt**. Comprueba el contenido del directorio y visualiza el contenido del fichero **miFich.txt**.
7. Visualiza el contenido del fichero **miFich.txt** usando los comandos **more** y **less**.
8. Muestra el número de líneas, palabras y caracteres del fichero **miFich.txt**.
9. Sitúate en **pruebasIIP** y mueve el directorio **prueba** al directorio **pract1**. Comprueba que el árbol de directorios de tu **DiscoW** es el que se muestra en la Figura 7.

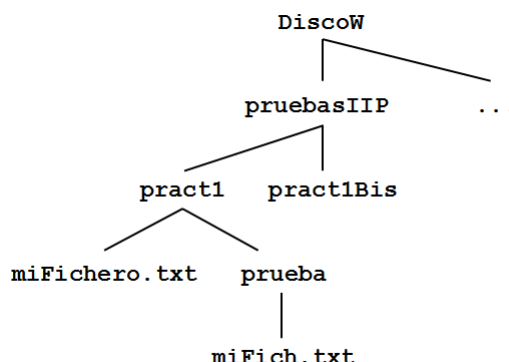


Figura 7: Árbol de directorios de tu DiscoW después de la actividad #8.

## 7. Edición, compilación y ejecución con BlueJ

*BlueJ* es un entorno de programación Java creado inicialmente por el equipo *BlueJ* de la Universidad de Monash de Merlbourne (Australia) y, actualmente, desarrollado y mantenido por un grupo de investigación de la Universidad de Kent en Canterbury y el King's College de Londres (Reino Unido), con el propósito específico de ser utilizado en cursos de introducción a la programación en Java.

A través de un interfaz gráfico bastante sencillo, *BlueJ* permite desarrollar cualquier aplicación Java. En *BlueJ* se denomina *proyecto* al directorio que contiene todos los ficheros que conforman una aplicación Java: los fuente (con la extensión `.java`) y los *bytecode* (con extensión `.class`).

Aunque la funcionalidad completa del entorno *BlueJ* se explicará con detalle en la práctica 2 y se irá conociendo a lo largo de las sesiones de laboratorio, se verá ahora cómo editar, compilar y ejecutar un programa Java.

*BlueJ* se puede invocar desde el menú desplegable del entorno gráfico del sistema (Aplicaciones - Programación - BlueJ 4.2.1) o bien desde la línea de comandos:

```
bluej &
```

Tras invocar a *BlueJ*, aparece automáticamente una ventana, *la ventana principal de BlueJ* (Figura 8(a)). Nótese que su zona central está vacía y en la parte superior se distingue un menú que permite acceder a la realización de ciertas operaciones sobre el proyecto (**Proyecto**), de edición (**Edición**); herramientas de *BlueJ* (**Herramientas**); de visualización (**Ver**) y de ayuda (**Ayuda**).

Para crear un nuevo proyecto *BlueJ* se selecciona la opción **Proyecto - Nuevo Proyecto** del menú y se le da un nombre.

Para crear una clase de un proyecto se hace clic en el botón **Nueva Clase...** y se le da un nombre y un tipo (en nuestros ejemplos siempre se elegirá **Class**). En la ventana principal de *BlueJ* aparece el icono de la clase creada (Figura 8(b)) que aparece rayado porque aún no ha sido compilada.

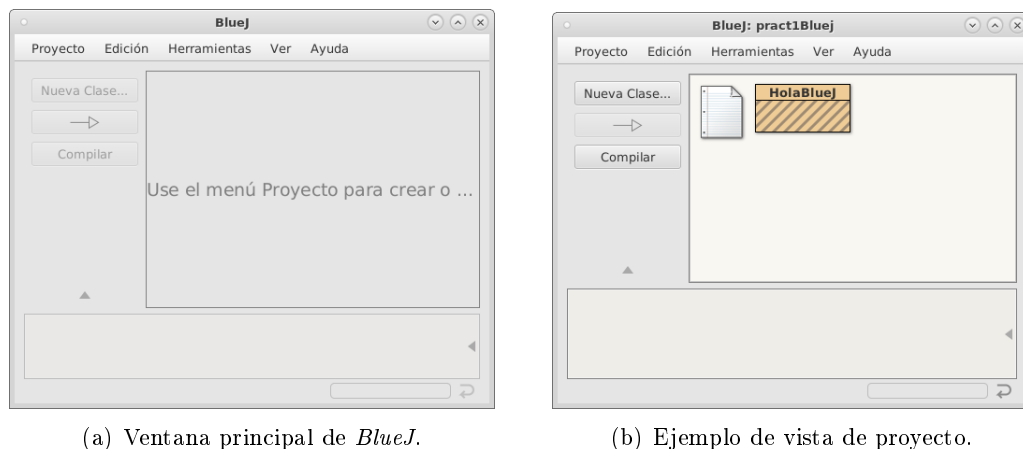


Figura 8: Ventanas de *BlueJ*.

Una vez creada una clase, al marcar su icono y hacer clic con el botón derecho del ratón, se obtienen las operaciones aplicables a la clase o *el menú de la clase* (Figura 9(a)). Para editar una clase se puede usar la opción **Abrir Editor** de dicho menú o también hacer doble clic sobre el icono de la clase.

En cuanto a cómo compilar una clase, existen varias maneras de hacerlo: desde el propio editor, desde el menú de la clase o desde el menú **Herramientas** de la ventana principal de *BlueJ*. Si el compilador detecta algún error, la línea que lo contiene aparece sombreada; así mismo, aparecerá un mensaje de error en *la zona de información* del editor, en la parte inferior de la pantalla (Figura 9(b)). Es más, si se pulsa en el enlace **Errors** que aparece en el extremo derecho de la zona de información del editor, se puede obtener más información acerca del error detectado.

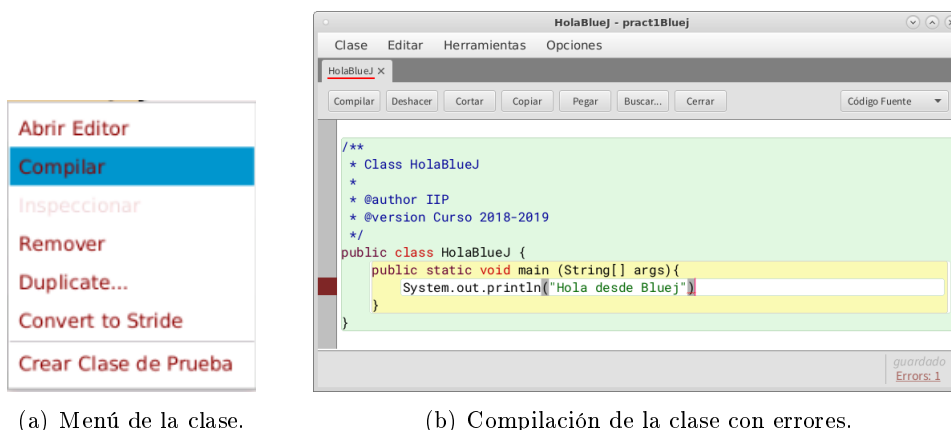


Figura 9: Compilación de la clase *HolaBlueJ*.

Aparte de los errores de compilación, el código debe escribirse siguiendo un estilo opcional (pero recomendado) que puede comprobarse desde la opción **Herramientas** -

**Checkstyle** en la vista de proyecto. Al elegir esta opción, aparece una nueva ventana donde las clases con errores de estilo se destacan en amarillo. Pinchando el nombre de la clase aparecen los avisos que indican las partes del código que no siguen el estilo predefinido que será usado durante la asignatura.

Como ya se ha mencionado anteriormente, al marcar el icono de una clase y hacer clic con el botón derecho del ratón, se obtiene el menú de la clase; pues bien, de la lista de operaciones que éste contiene destacaremos ahora la operación de llamada al método **main** de la clase; *se puede ejecutar una clase desde su menú* (Figura 10(a)).

En la Figura 10(b) se muestra la ventana de la llamada al método **main** de la clase **HolaBlueJ**. Si el programa solicita datos desde el teclado o muestra por pantalla algún resultado, aparece de forma automática un terminal de texto, la *ventana de terminal* (Figura 10(c)). Si no aparece, se debe seleccionar **Mostrar Terminal** del menú **Ver**.

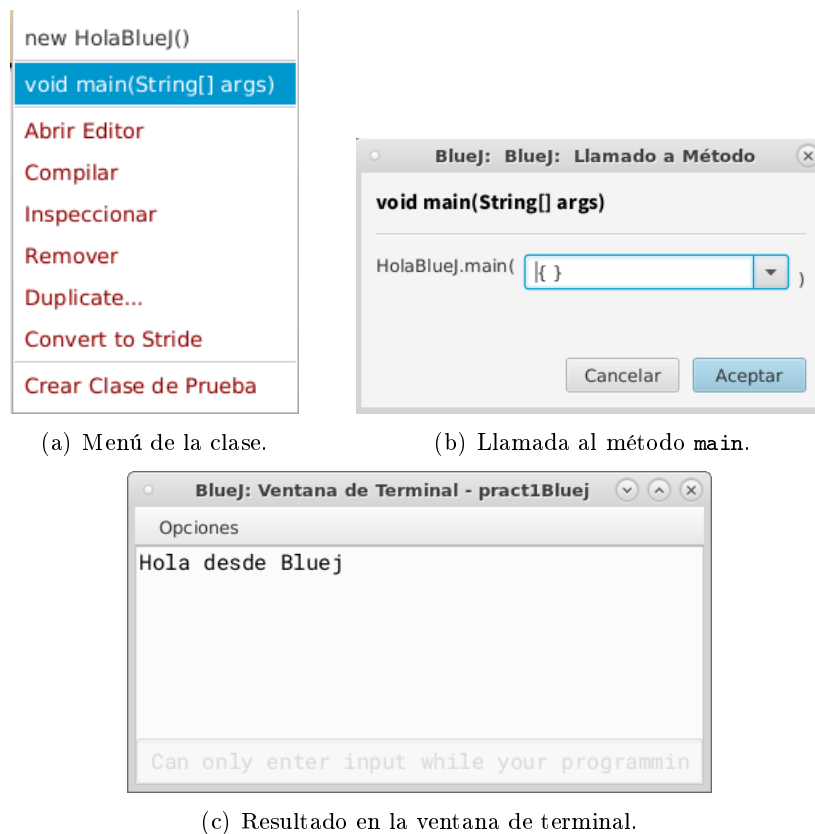


Figura 10: Ejecución de la clase **HolaBlueJ**.

## Actividad #9

1. Situado en **pruebasIIP**, llama a *BlueJ* sin parámetros y crea el proyecto (a todos los efectos un directorio) **pract1Bluej** a través de la opción **Proyecto - Nuevo Proyecto**

2. Crea la clase `HolaBlueJ` y abre el editor para introducir el siguiente código:

```
/** Clase HolaBlueJ */
public class HolaBlueJ {
    public static void main(String[] args) {
        System.out.println("Hola desde Bluej");
    }
}
```

3. Compila la clase, corrige los errores de compilación y comprueba qué ocurre en la parte central de la ventana principal de *BlueJ*.
4. Comprueba el estilo y corrige los posibles errores de estilo.
5. Ejecuta el método `main` de la clase `HolaBlueJ` y comprueba que el resultado es el que se muestra en la Figura 10(c). Puedes guardar dicho resultado en un fichero de texto con la opción **Opciones - Guardar en archivo** del menú de la ventana de terminal.
6. Fíjate que el árbol de directorios ahora tiene el aspecto de la Figura 11.

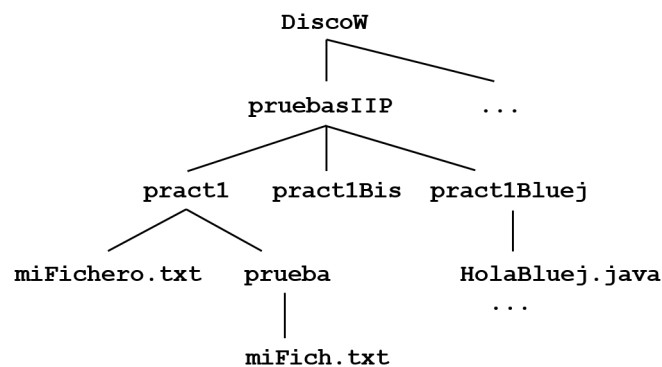


Figura 11: Árbol de directorios de tu `DiscoW` después de la actividad #9.

**NOTA:** Cuando acabes la práctica cierra la sesión, **NO APAGUES** la computadora. Desde el menú **Sistema** elige **Cerrar la sesión de ...**. Aparecerá una ventana en la que, si pulsas el botón **Cerrar la sesión**, continua el proceso de desconexión y, al salir de la sesión MATE, vuelve a aparecer la pantalla inicial de conexión donde se pide el login y el password.



## 8. Anexos

### 8.1. Otros comandos útiles en Linux

#### Redireccionamiento de entrada y salida

En la mayoría de los comandos se introducen datos desde teclado (*entrada estándar*), se realiza un proceso con esos datos y se obtiene una salida en pantalla (*salida estándar*). A continuación se muestra cómo redirigir la entrada/salida estándar de un comando desde/a un fichero, respectivamente.

- **comando > fichDestino** redirecciona la salida estándar del comando al fichero **fichDestino**. Si dicho fichero no existe, se crea y si ya existe, se reemplaza su contenido.
- **comando >> fichDestino** es igual que el anterior, pero añadiendo la información al final del fichero.
- **comando 2> fichOrigen** redirecciona la salida de error del comando al fichero **fichOrigen**.
- **comando < fichOrigen** redirecciona la entrada estándar del comando al fichero **fichOrigen**.
- **comando<sub>1</sub> | comando<sub>2</sub>** redirecciona la salida estándar del primer comando a la entrada estándar del segundo comando.
- **tee fichDestino<sub>1</sub> ... fichDestino<sub>n</sub>** copia su entrada estándar en su salida estándar y en todos los ficheros pasados como argumentos. El conducto **|** combinado con el comando **tee** permite obtener un fichero de la ejecución de un programa al tiempo que se visualiza en la pantalla.

#### Control de acceso, seguridad y privilegios

Los siguientes comandos permiten cambiar la clave de acceso al sistema y el modo de acceso a un fichero (quién y qué operaciones se pueden realizar con el fichero).

- **passwd** cambia la contraseña.
- **chmod modo nombre** cambia los permisos de acceso al fichero o directorio **nombre** siguiendo el **modo**, pudiendo ser este último, bien una representación de los cambios a realizar, bien un número en octal con el patrón de bits de los nuevos permisos.

#### Gestión básica del sistema

Otros comandos para la gestión básica del sistema son los que siguen:

- **who** lista los usuarios actuales del sistema.
- **whoami** lista el nombre del usuario que lanza el comando.
- **&** escrito detrás de un comando, hace que el mismo se ejecute concurrentemente en un segundo plano, sin dejar bloqueado el terminal mientras dure su ejecución.
- **ps** lista los procesos activos.

- `kill -9 PID` termina inmediatamente la ejecución del proceso con código PID.
- `env` muestra el entorno del usuario permitiendo modificarlo.
- `exit` finaliza la sesión en curso (cierra el terminal).

## 8.2. Acceso remoto a los laboratorios

Se puede acceder remotamente a los laboratorios del DSIC (tanto en modo Linux como Windows) usando alguna utilidad de escritorio remoto. Con ello, puedes trabajar desde cualquier ordenador que tenga acceso a Internet como si estuvieras ante un terminal de los laboratorios. El acceso remoto a los laboratorios está garantizado durante el horario docente, tal y como se menciona en los términos de uso del servicio.

En la url <http://www.upv.es/entidades/DSIC/index>, enlace “Escritorios virtuales (EVIR)”, están descritos los programas y parámetros básicos para hacerlo desde cualquiera de los sistemas soportados: Linux, Mac OS X y Windows.

Su uso es sencillo, aunque es conveniente tener una conexión rápida a Internet (suele ser suficiente con una de las ADSLs habituales hoy en día). Además, al iniciar la sesión remota tendrás que identificarte con tu nombre de usuario y contraseña usados en los laboratorios del DSIC.

Como ejemplo, puedes acceder desde un sistema Windows a una sesión de trabajo en Linux con un escritorio y recursos idénticos a los que encontrarás al iniciar una sesión en el laboratorio. Igualmente, es posible acceder a una sesión Windows remota desde un Mac, Windows o Linux con el soporte adecuado instalado.

El DSIC recomienda la utilización de estos recursos y proporciona además una guía escrita básica para su instalación y uso.