

Programació I

Primers contactes amb la programació: edició, compilació i execució de codi Java.

Facultat de Matemàtiques i Informàtica

Grau d'Enginyeria Informàtica

22 de setembre de 2020



Índex

1	Introducció	2
2	Passos en la realització d'un programa: edició, compilació, depuració i execució.	2
3	Programar Java a Moodle (Caseine)	3
4	Programar Java des d'un IDE	4
4.1	Introducció	4
4.2	Instal·lació de l'IDE Netbeans	4
4.3	Crear un nou projecte Netbeans: edició, compilació i execució	5
4.4	Creació d'un projecte a Netbeans a partir d'un fitxer .java	6
5	Programar Java des de consola	8
5.1	Comandes bàsiques de consola	8
5.2	Edició d'un fitxer font Java	8
5.2.1	Editor visual (Kwrite)	9
5.2.2	Editor en consola (vim)	9
5.3	Compilació i execució	10

1 Introducció

En aquest manual aprendreu els passos a seguir per a crear un programa d'ordinador, en general, i en particular amb el llenguatge de programació Java. Concretament, aprendreu a engegar les utilitats on podeu programar en Java i realitzareu el vostre primer programa.

2 Passos en la realització d'un programa: edició, compilació, depuració i execució.

Els passos principals en la realització d'un programa són: **editar** del programa (pensar què escriure i escriure'l), **compilar-lo** (principalment, comprovar que està ben escrit), **depurar-lo** (arreglar errors, si n'hi ha) i **executar-lo** (comprovar que fa el que ha de fer). Penseu en l'analogia del procés que un escriptor segueix per a escriure una novel·la. Primer, l'escriptor pensa els personatges, el lloc on es desenvolupa la història, la trama, etc. Després, es posa a escriure cadascuns dels capítols de la novel·la. Més tard, el corrector, repassa l'esborrany del llibre. Si el corrector troba paràgrafs que no s'entenen o ha trobat errors ortogràfics, informa l'escriptor perquè ho revisi. Finalment, s'imprimeix el llibre i es porta a la botiga per a que el puguin comprar els lectors.

La Figura 1 mostra els passos que s'han de realitzar per a crear un programa Java: editar el codi font (fitxer amb extensió .java), compilar el codi font per a obtenir el codi intermedi (es crea automàticament el fitxer .class, si no hi ha errors al codi font) i finalment executar el programa. Més endavant aprendreu a **depurar** codi (a fer "debugging"). Depurar codi és el procés de buscar i trobar errors als nostres programes. Val a dir què aquests passos són els que es realitzen a altres llenguatges com ara C, C++ (llenguatges compilats). A llenguatges interpretats com python, fareu l'edició de codi i a continuació l'execució del mateix.

En aquest document expliquem com programar a 3 entorns diferents: i) a Caseine (Moodle), ii) a un IDE (Integrated Development Enviroment) com Netbeans, i finalment iii) fent ús de la consola/terminal de Linux/Mac/Windows.

La nostra recomanació per alumnes que mai han programat és fer servir Caseine els primers dies/-setmanes (no és necessari fer instal·lacions a l'ordinador de casa), després, si voleu, podeu fer servir Netbeans. També podeu fer servir la consola de comandos, però encara que és molt ràpid l'edició, compilació i execució de codi, la depuració es fa més feixuga.

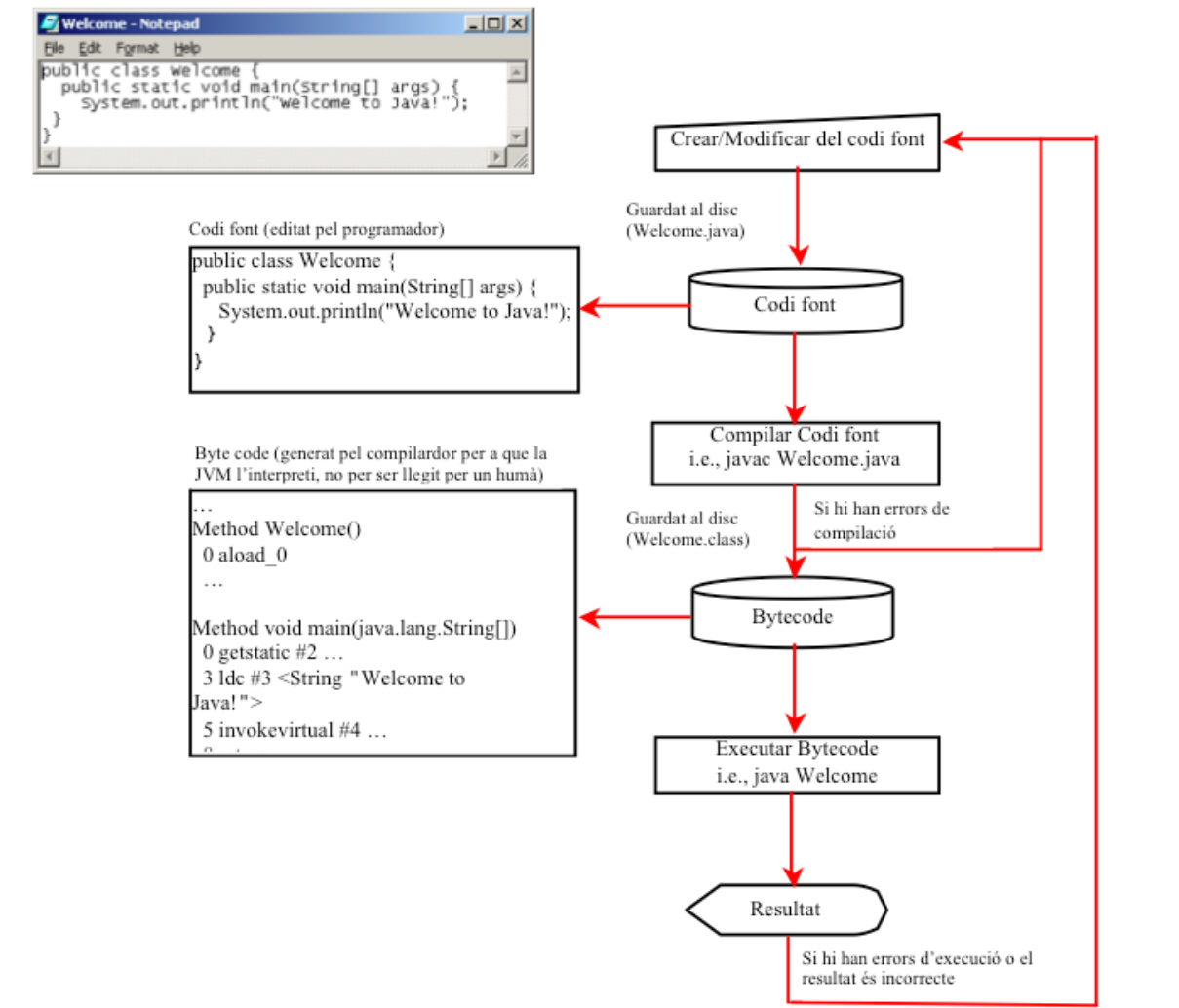


Figura 1: Passos per a crear un programa Java

3 Programar Java a Moodle (Caseine)

Moodle és un LMS (Learning Management System) que permet utilitzar recursos del tipus Virtual Programming Lab (VPL). El professor pot crear recursos VPL al curs on podreu editar, compilar i executar codi Java. Veure la Figura 2 per a saber com utilitzar un recurs VPL a Caseine, només heu de (1. Entrar al recurs) clicar el link del recurs, i una vegada dintre (2. Edit view) heu d'editar el codi, executar i avaluar (testejar). Si al compilar hi ha errors de compilació, heu de arreglar-los i tornar a clicar "Executar". També podeu veure els (3. Submission view) resultats del que heu fet.

També, com veurem més endavant al llarg del curs, podreu validar (comprovar que funciona correctament) el vostre programa fent ús de proves unitàries (JUnit).

Val a dir que moltes de les activitats VPL que trobareu al campus de l'assignatura a Caseine són d'autoaprenentatge, la nota és orientativa per a vosaltres mateixos saber com heu assolit els coneixements

1. Entrar al recurs

Conec els passos necessaris per a crear un programa.

Llegir sobre errors de compilació i errors d'execució.

Sé diferenciar, i arreglar, errors de compilació i d'execució.

Fes el teu primer programa: HelloWorld (6)

HelloWorld amb dades d'entrada: HelloWorldInput (6)

3. Submission view

Home / My courses / Progl_UB / Setmana 1: 28 set - 2 oct 2020 / Fes el teu primer programa

Description Submission Edit Submission view Previous submissions list

Grade

Reviewed on Tuesday, 22 September

Grade: 7.00 / 10.00

Final reduction: 3 [5 / 2 -1]

Assessment report [1]

Summary of tests

Submitted on Tuesday, 22 September

HelloWorld.java

```
1 public class HelloWorld{
```

2. Edit view

Description Submission Edit Submission view Previous submissions list

```
1 public class HelloWorld{
2
3
4
5 //TODO
6 //1) Compila i executa el programa
7 //2) Modifica el programa per a mostrar
8 // per pantalla el missatge "Hello world!!";
9 System.out.println("Hello World!!");
10
11 }
```

Proposed grade: 7 / 10

Summary of tests

3 test run / 3 test passed

Avaluar

Executar

Testeja el codi fent servir proves definides pel professor/a

"Executar" fa ambdues coses: primer **compila** i, si no hi ha errors, **executa**

- Quan compila, crida internament al compilador de java (javac)
- Quan executa, crida internament a la java virtual machine (jvm) (java)

Recordeu, si hi ha errors de compilació o execució, s'han d'arreglar els errors, desmarcar els canvis i tornar a clicar "Executar"!

Figura 2: Instruccions per a utilitzar recursos de programació, VPL, a Caseine

i habilitats que es treballen en aquesta activitat.

4 Programar Java des d'un IDE

4.1 Introducció

IDE significa Integrated Development Environment, dit d'altre forma, un IDE és un entorn on podeu editar, compilar i executar programes. Concretament, Netbeans és un entorn de desenvolupament lliure fet amb Java. Altre IDE molt conegut és Eclipse.

4.2 Instal·lació de l'IDE Netbeans

Als ordinadors del laboratori ja tenim instal·lat Netbeans, per instal·lar Netbeans als vostres ordinadors haureu d'assegurar-vos d'instal·lar conjuntament l'anomenat JDK (Java SE Development Kit). A la pàgina d'apache podeu baixar l'última versió de l'IDE (aneu on diu Installers i seleccioneu el fitxer que correspon amb el vostre sistema operatiu). <https://netbeans.apache.org/download/nb121/nb121.html> L'instal·lador us guiarà en el procés. Una vegada instal·lat, per obrir "apache Netbeans", podeu buscar-lo al llançador d'aplicacions.

A vegades creareu un programa Java des de 0, doncs voldreu crear un nou projecte Netbeans. Unes altres ja tindreu un (o varis) fitxers .java i voldreu crear un projecte Netbeans per afegir aquests fitxers. A continuació podeu trobar com fer-ho.

4.3 Crear un nou projecte Netbeans: edició, compilació i execució

A la barra de tasques de Netbeans feu 'File → New Project' (veure Figura 3). S'obrirà un menú contextual que demana per la configuració del projecte. Escolliu 'Java Application' i 'Next' (veure Figura 4).

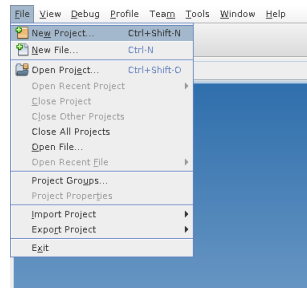


Figura 3: Creació d'un projecte a Netbeans

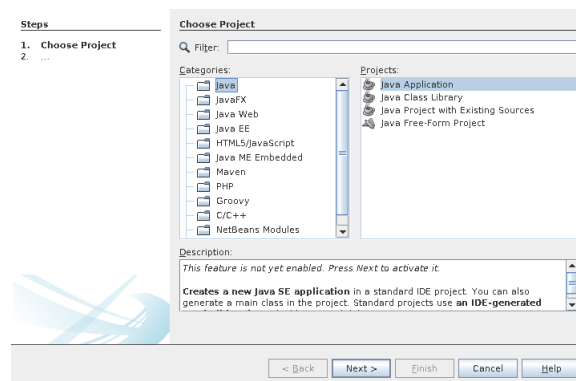


Figura 4: Creació d'un projecte a Netbeans pas 1

Doneu nom al projecte, a l'exemple el nom és 'HelloWorld2' i finalitzeu. Netbeans generarà la classe principal i el mètode main() (veure Figura 5).

Dins del main() escriurem la sentència:

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

que farà que s'escrigui per la consola de Netbeans el conjunt de caràcters "Hello World!".

A continuació, per compilar i executar a Netbeans hem de clicar a la fletxa verda de la barra d'eines o pitjar **F6** (veure Figura 6). Si no hi ha errors de compilació, s'executarà directament. A la part inferior de la finestra ha d'apareixi el missatge "Hello World!". Si hi ha errors (per exemple, falta un ; o no s'ha tancat una clau), s'han de corregir i tornar a clicar la fletxa verda.

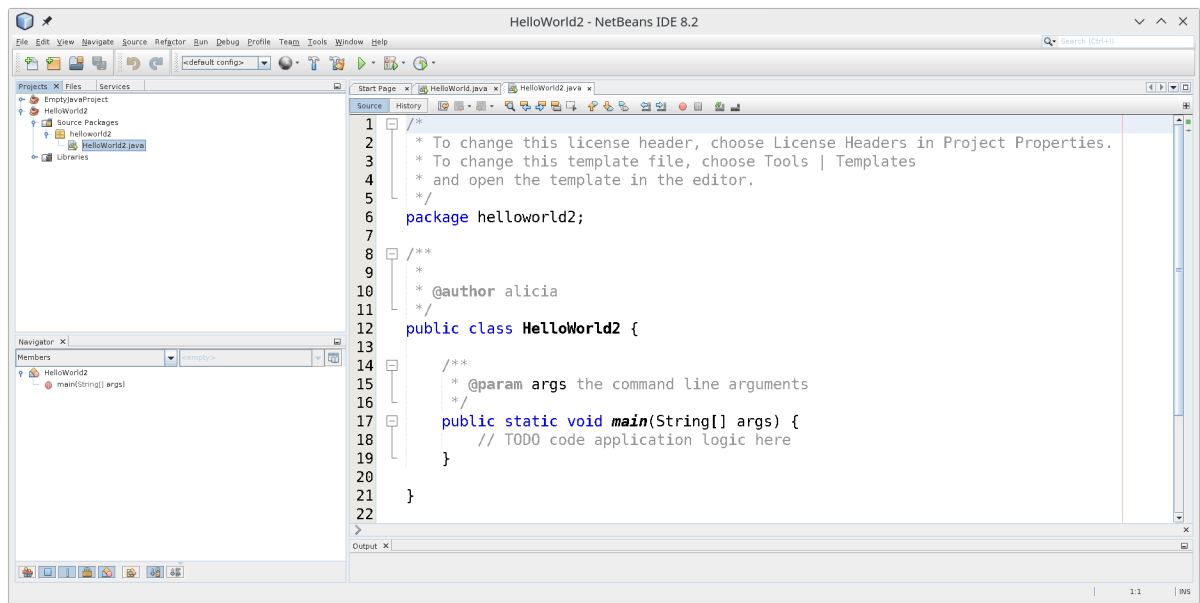


Figura 5: IDE Netbeans amb classe i mètode main

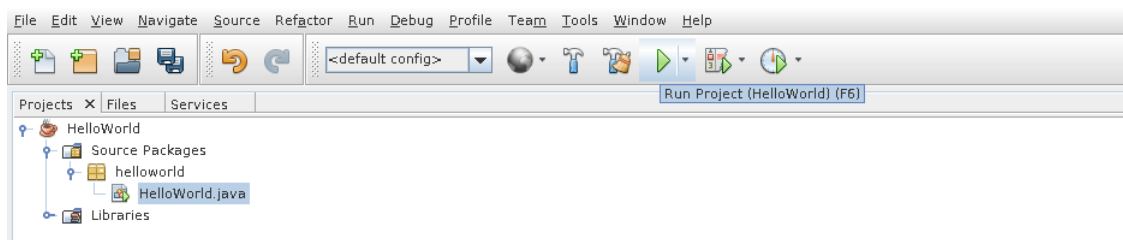


Figura 6: Execució d'un projecte a Netbeans

4.4 Creació d'un projecte a Netbeans a partir d'un fitxer .java

Si el que volem és crear un projecte Netbeans a partir d'un fitxer Java ja existent, heu de crear un nou projecte de manera similar al apartat anterior excepte que a l'hora de configurar el projecte al pas 2 (Figura 7) desmarqueu la casella que pregunta per la creació automàtica de la classe principal.

Tot seguit només caldrà que arrossegeu l'arxiu que volem afegir a la carpeta source del projecte Netbeans (veure Figure 8).

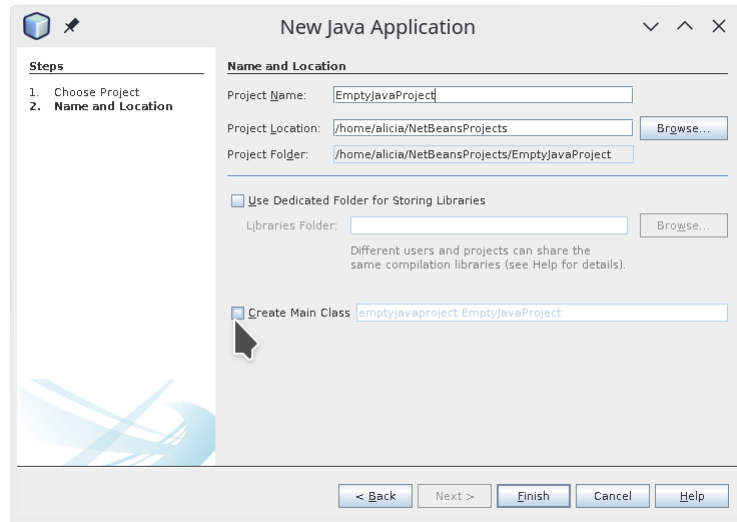


Figura 7: Desmarcar casella 'Create Main Class', projecte de Netbeans a partir d'un fitxer .java existent

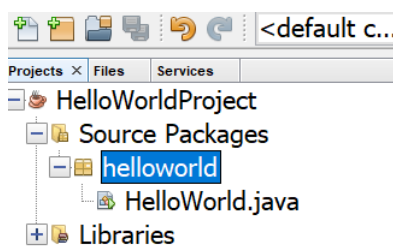



Figura 8: Estructura de carpetes del projecte Netbeans

5 Programar Java des de consola

La consola o terminal és una finestra que permet teclejar comandes i executar programes. Una vegada carregat el sistema operatiu Linux (<https://ca.wikipedia.org/wiki/Linux>) podeu obrir la consola de diverses formes, una d'elles és fer clic al Dash button (logo de l'escriptori, normalment abaix a l'esquerra), escriure "Terminal" a la barra de cerca i . Podeu veure que per defecte a Linux us situeu al vostre directori home (/home/vostre-usuari) (veure Figura 9) .

5.1 Comandes bàsiques de consola

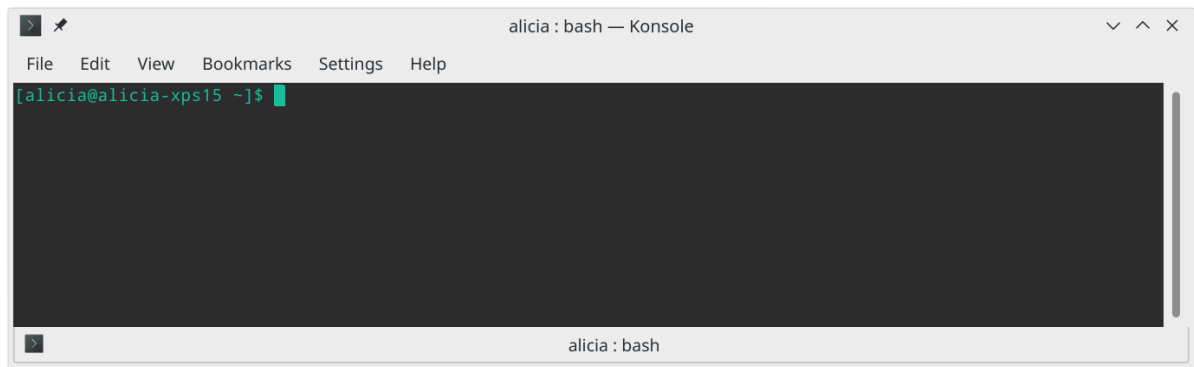


Figura 9: Finestra de consola

Un cop dins la consola veiem les principals comandes per mouren's a través dels directoris:

- **ls**: mostra el contingut del directori actual (amb **ls -l** es mostren més detalls).
- **cd_nomCarpeta**: duu dins la carpeta amb nom *nomCarpeta*.
- **cd_..**: duu al directori superior de l'actual.
- **mkdir_nomCarpeta**: crea un directori amb nom *nomCarpeta*.
- **pwd**: mostra el directori actual. Quan s'obre la consola, **pwd** mostra el directori home de l'usuari.
- **rm_nomFitxer**: esborra el fitxer amb nom *nomFitxer*.
- **cat_nomFitxer**: mostra el contingut del fitxer amb nom *nomFitxer*.
- **cp_nomFitxerOrig_nomFitxerCopia**: copia el fitxer *nomFitxerOrig* a *nomFitxerCopia*.
- **man_nomComando**: mostra informació sobre el comando *nomComando*.


5.2 Edició d'un fitxer font Java

Respecte a l'edició d'un fitxer en codi font Java, aquesta s'ha de fer en un editor de text pla (com ara Notepad en Windows, o Kwrite i vim en Linux).

A continuació podeu veure com fer-ho amb l'editor visual *Kwrite* i amb l'editor de consola *vim*. A casa, com s'ha dit abans, podeu fer servir Netbeans per a tot (edició, compilació i execució).

5.2.1 Editor visual (Kwrite)

Kwrite és un editor de text pla que podeu trobar a les aules. Mitjançant Kwrite podeu escriure el codi Java que després compilareu i executeu per consola (veure Figura 10). Per obrir Kwrite per consola:

`kwrite_HelloWorld.java` 

i s'obrirà la finestra d'edició de Kwrite editant el fitxer *HelloWorld.java*.

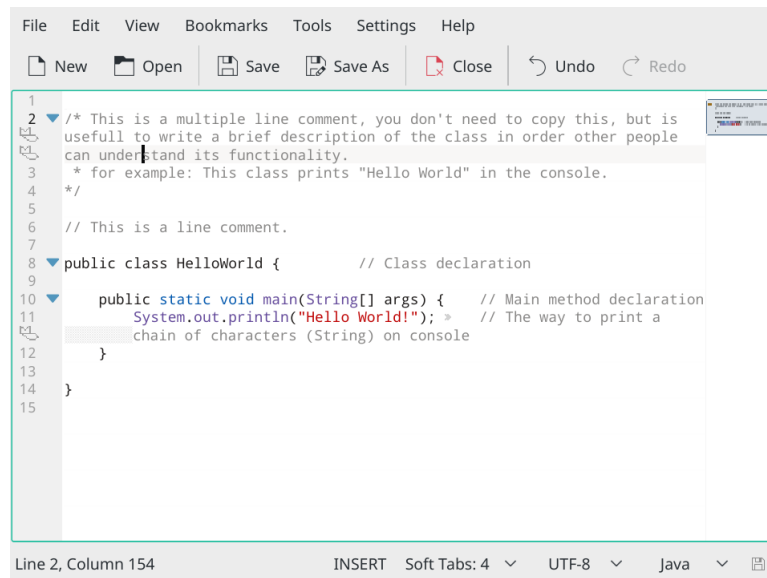


Figura 10: Kwrite amb codi Java de programa Hello World

Una vegada acabada l'edició del fitxer, deseu l'arxiu prement "Save" de la barra d'eines. Fixeu-vos bé on esteu desant el fitxer, després heu d'anar a aquest directori des de la consola per compilar i executar el programa Java.

5.2.2 Editor en consola (vim)

Vim és un editor de text pla que s'executa en consola (Figura 11), si no s'hi està familiaritzat, pot resultar un pèl més complicat però és interessant conèixer la seva existència, ja que una vegada après pot agilitzar molt l'edició respecte a l'ús d'una GUI (Graphical User Interface). Per obrir vim, podeu teclejar a la consola:

- `vim_HelloWorld.java`: comença a editar amb Vim un nou arxiu anomenat *HelloWorld.java*

A l'última línia visible de la consola Vim ens mostra el mode en què ens trobem a l'arxiu. Modes de Vim:

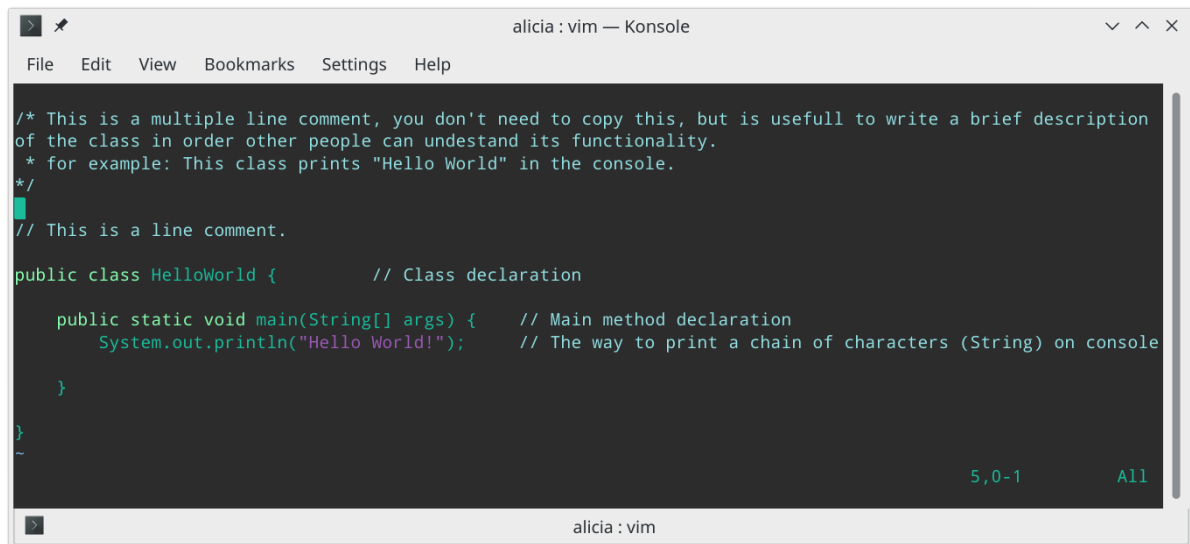


Figura 11: HelloWorld.java editant a Vim a mode normal

- **Mode normal:** Aquest és el mode en el que el vim comiença la seva execució. Per a indicar el mode, Vim no mostra res o el nom de l'arxiu.
- **Mode inserció:** És el mode edició, en el que s'inserta text. Per a indicar el mode, Vim mostra `--INSERT--`. Es pot passar a mode normal teclejant ESC.

Per tal de començar a editar heu de ser al mode inserció. Podeu passar de mode normal al mode inserció pitjant la tecla `i` al teclat. Ja podreu escriure el codi a Vim. Després de l'edició torneu al mode normal pitjant `[Esc]` i deseu l'arxiu escrivint a consola de Vim: `w` i `:wq` per desar i sortir.

5.3 Compilació i execució

Rere haver editat i desat l'arxiu de codi Java podeu fer `ls` a la consola i observar l'arxiu creat al directori. Mitjançant la instrucció:

```
javac HelloWorld.java
```

I sempre que no hi hagi errors de compilació, es genera l'arxiu compilat `.class`. En el cas de trobar un error (Figura 12) apareixerà un missatge indicant de quin error es tracta i a quina línia del codi hi és. Haureu doncs de tornar a obrir el fitxer amb l'editor i corregir l'error.

Un cop compilat el programa sense errors, ja podeu executar el programa amb (veure Figura 13):

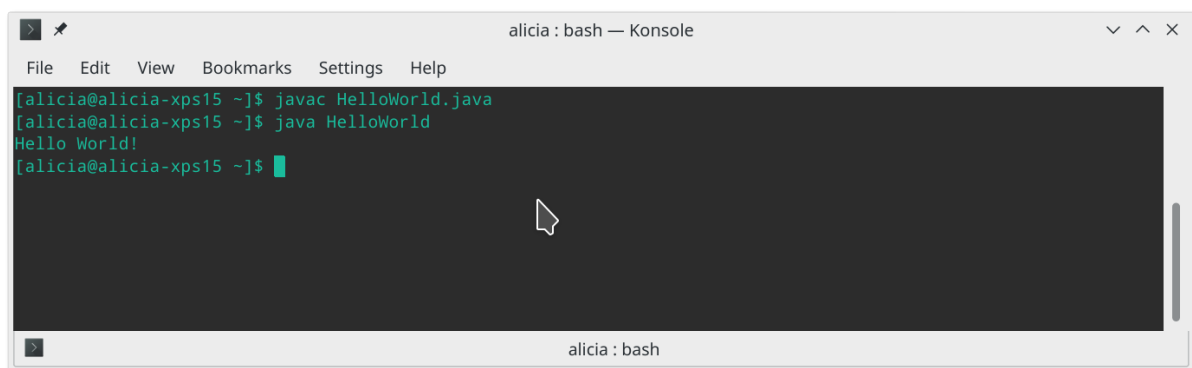
```
java HelloWorld
```



The screenshot shows a terminal window titled "alicia : bash — Konsole". The terminal content is as follows:

```
[alicia@alicia-xps15 ~]$ javac HelloWorld.java
HelloWorld.java:11: error: ';' expected
    System.out.println("Hello World!")    // The way to print a chain of characters (String) on console
                                ^
1 error
[alicia@alicia-xps15 ~]$
```

Figura 12: Error de compilació mostrat per consola



The screenshot shows the same terminal window after successful compilation and execution. The terminal content is as follows:

```
[alicia@alicia-xps15 ~]$ javac HelloWorld.java
[alicia@alicia-xps15 ~]$ java HelloWorld
Hello World!
[alicia@alicia-xps15 ~]$
```

Figura 13: HelloWorld executat