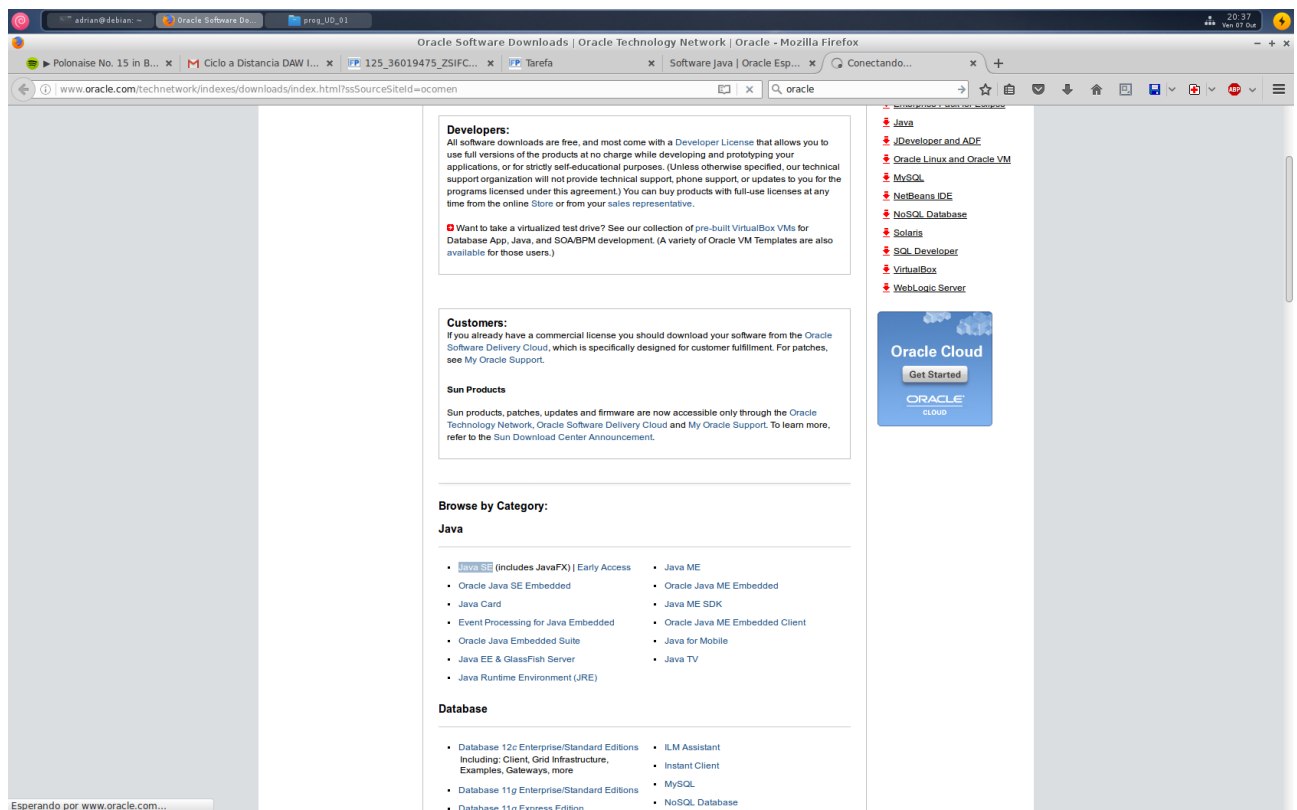


Alumno: Adrián Álvarez Lois
PROG_UD_01

Apartado A:

1. Descarga desde la web recomendada en los contenidos Java SE e instalación en mi equipo.



Java SE Development Kit 8 - Downloads - Mozilla Firefox

Polonaise No. 15 in B... x Ciclo a Distancia DAW I... x 125_36019475_Z5IFC... x PP: Tarefa x Software Java | Oracle Esp... x Java SE Development ... x

www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html

Java EE
Java ME
Java SE Support
Java SE Advanced & Suite
Java Embedded
Java DB
Web Tier
Java Card
Java TV
New to Java
Community
Java Magazine

Java SE Development Kit 8 Downloads

Thank you for downloading this release of the Java™ Platform, Standard Edition Development Kit (JDK™). The JDK is a development environment for building applications, applets, and components using the Java programming language.

The JDK includes tools useful for developing and testing programs written in the Java programming language and running on the Java platform.

See also:

- Java Developer Newsletter: From your Oracle account, select **Subscriptions**, expand **Technology**, and subscribe to **Java**.
- Java Developer Day hands-on workshops (free) and other events
- Java Magazine

JDK 8u101 Checksum
JDK 8u102 Checksum

Java SE Development Kit 8u101

You must accept the **Oracle Binary Code License Agreement for Java SE** to download this software.

☐ Accept License Agreement ☒ Decline License Agreement

Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	77.77 MB	jdk-8u101-linux-arm32-vfp-hflt.tar.gz
Linux ARM 64 Hard Float ABI	74.72 MB	jdk-8u101-linux-arm64-vfp-hflt.tar.gz
Linux x86	160.28 MB	jdk-8u101-linux-i586.rpm
Linux x86	174.96 MB	jdk-8u101-linux-i586.tar.gz
Linux x64	158.27 MB	jdk-8u101-linux-x64.rpm
Linux x64	172.95 MB	jdk-8u101-linux-x64.tar.gz
Mac OS X	227.36 MB	jdk-8u101-macosx-x64.dmg
Solaris SPARC 64-bit	139.66 MB	jdk-8u101-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	98.96 MB	jdk-8u101-solaris-sparcv9.tar.gz
Solaris x64	140.33 MB	jdk-8u101-solaris-x64.tar.Z
Solaris x64	96.78 MB	jdk-8u101-solaris-x64.tar.gz
Windows x86	188.32 MB	jdk-8u101-windows-i586.exe
Windows x64	193.68 MB	jdk-8u101-windows-x64.exe

Java SE Development Kit 8u102

You must accept the **Oracle Binary Code License Agreement for Java SE** to download this software.

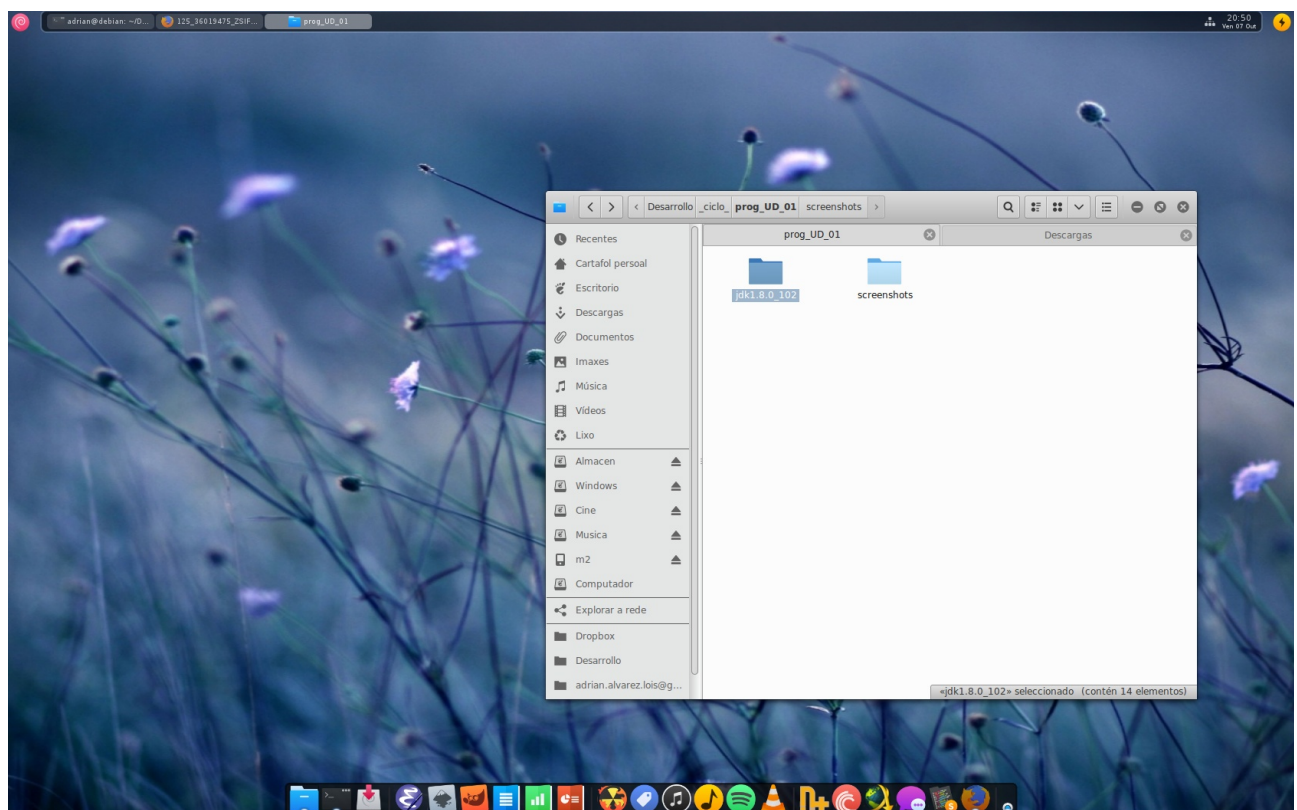
☐ Accept License Agreement ☒ Decline License Agreement

Product / File Description	File Size	Download
Linux x86	160.35 MB	jdk-8u102-linux-i586.rpm
Linux x86	175.03 MB	jdk-8u102-linux-i586.tar.gz
Linux x64	158.35 MB	jdk-8u102-linux-x64.rpm
Linux x64	173.03 MB	jdk-8u102-linux-x64.tar.gz
Mac OS X	227.35 MB	jdk-8u102-macosx-x64.dmg
Solaris SPARC 64-bit	139.59 MB	jdk-8u102-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	98.98 MB	jdk-8u102-solaris-sparcv9.tar.gz
Solaris x64	140.02 MB	jdk-8u102-solaris-x64.tar.Z
Solaris x64	96.24 MB	jdk-8u102-solaris-x64.tar.gz
Windows x86	189.2 MB	jdk-8u102-windows-i586.exe
Windows x64	194.68 MB	jdk-8u102-windows-x64.exe

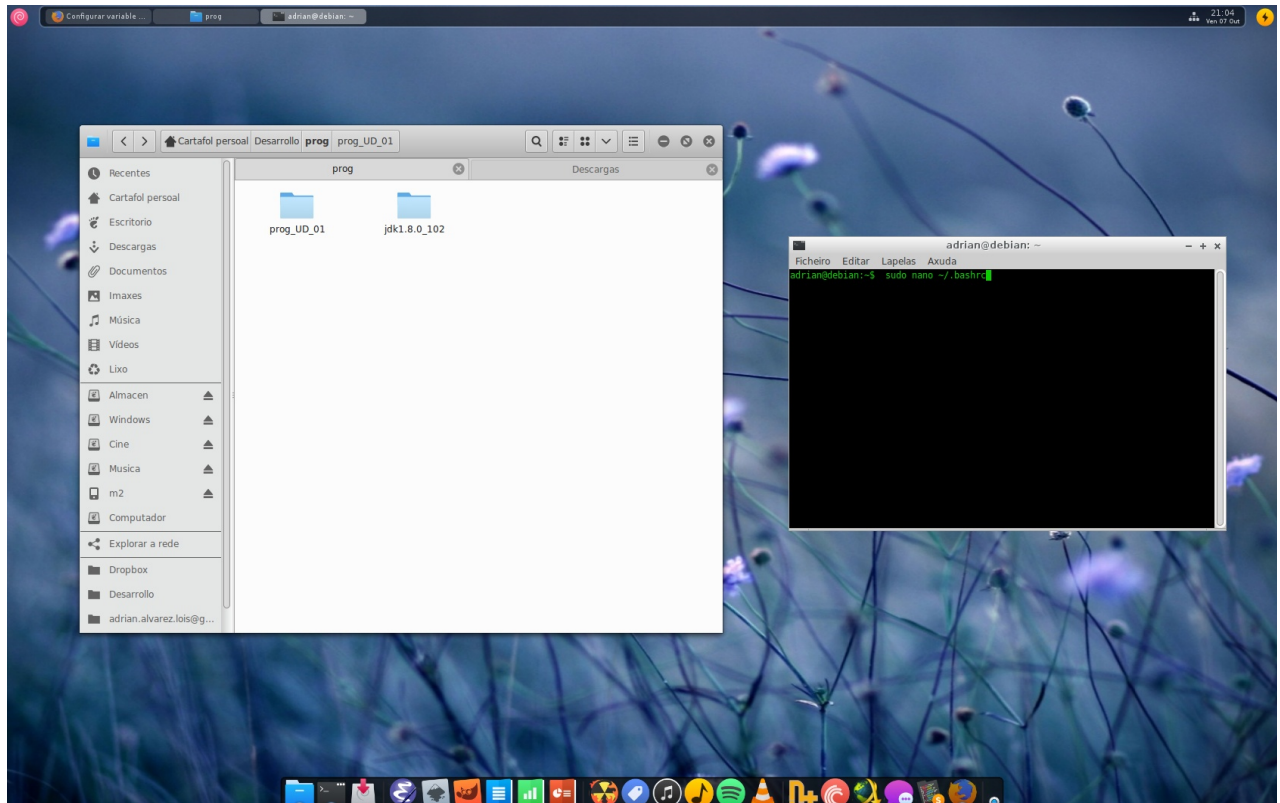
Java SE Development Kit 8u101 Demos and Samples Downloads

You must accept the **Oracle BSD License**, to download this software.

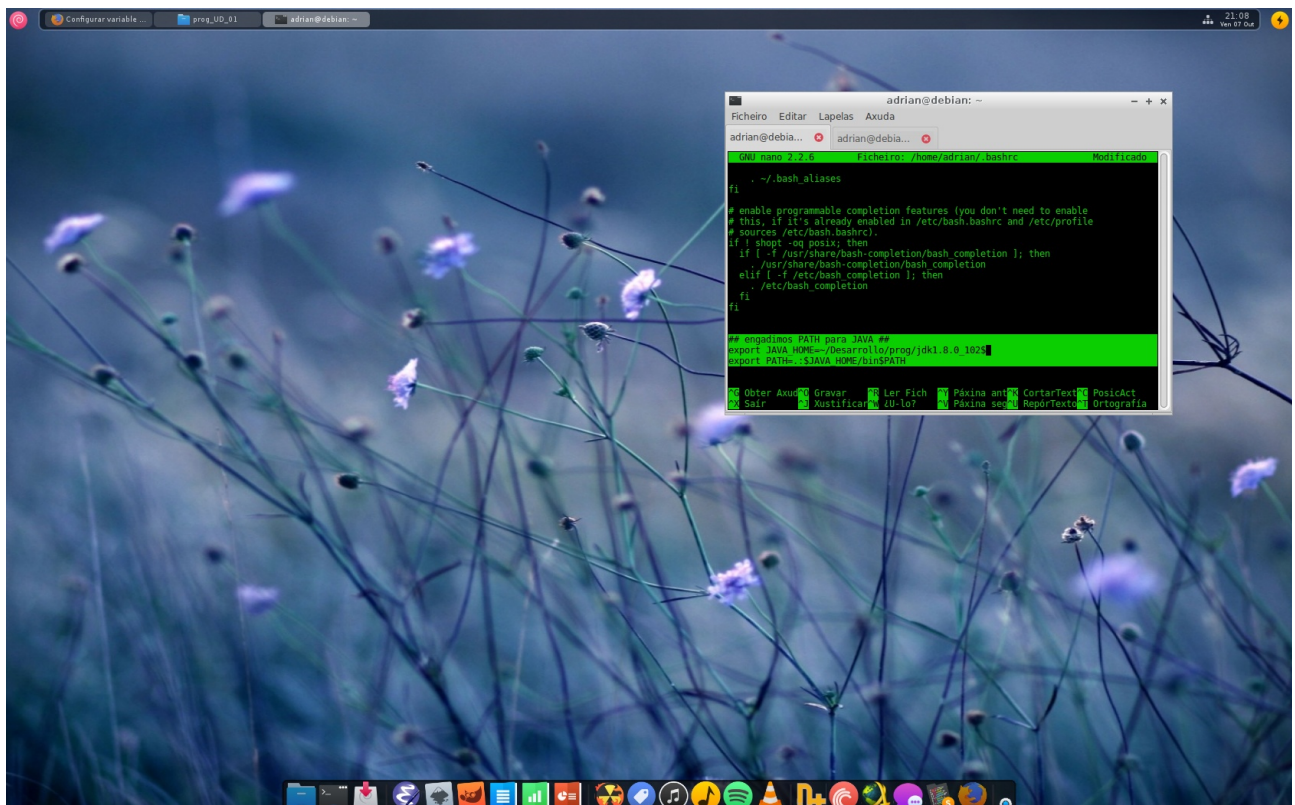
Java SE
Java EE and Glassfish
Java ME
Java Card
NetBeans IDE
Java Mission Control
Java Resources
Java APIs
Technical Articles
Demos and Videos
Forums
Java Magazine
Java.net
Developer Training
Tutorials
Java.com



2. Configurar las variables PATH y CLASSPATH.

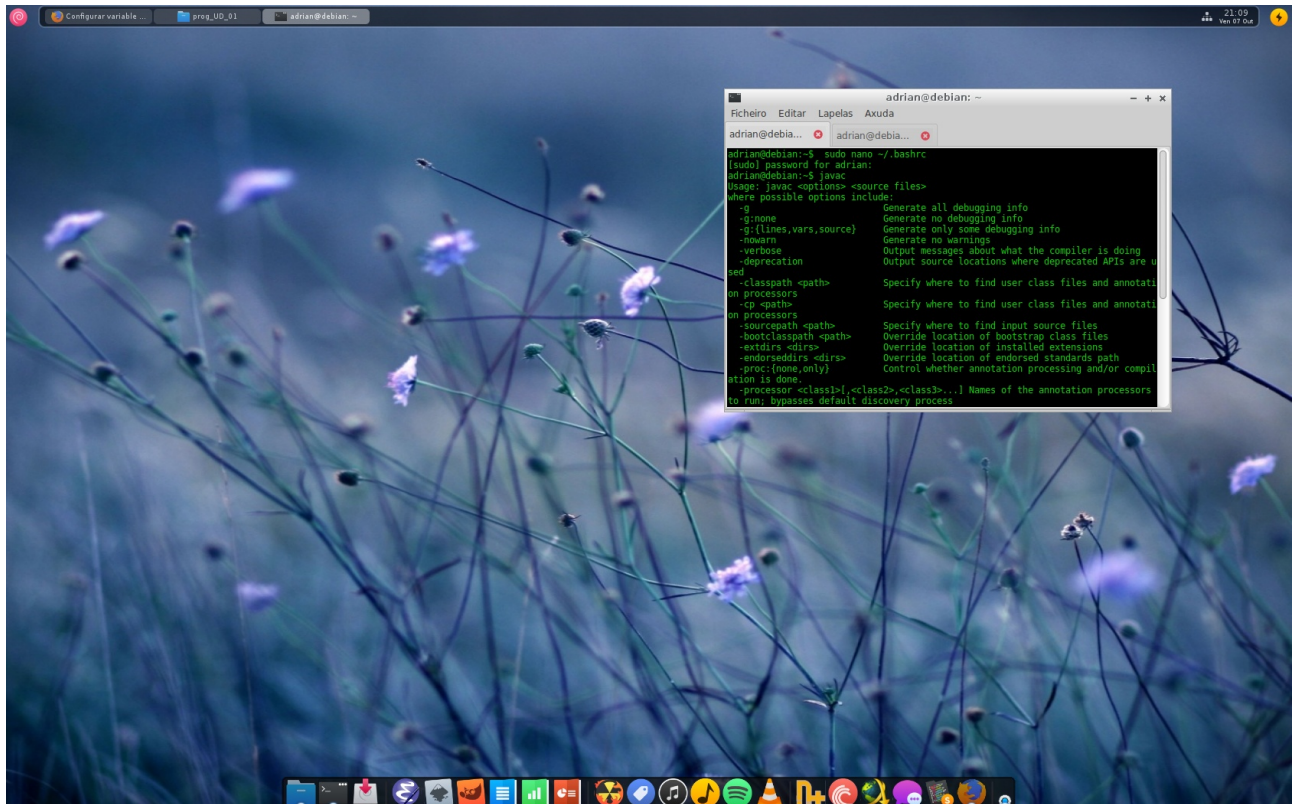


En mi Debian le digo al sistema donde se encuentran los binarios editando los parámetros del terminal.



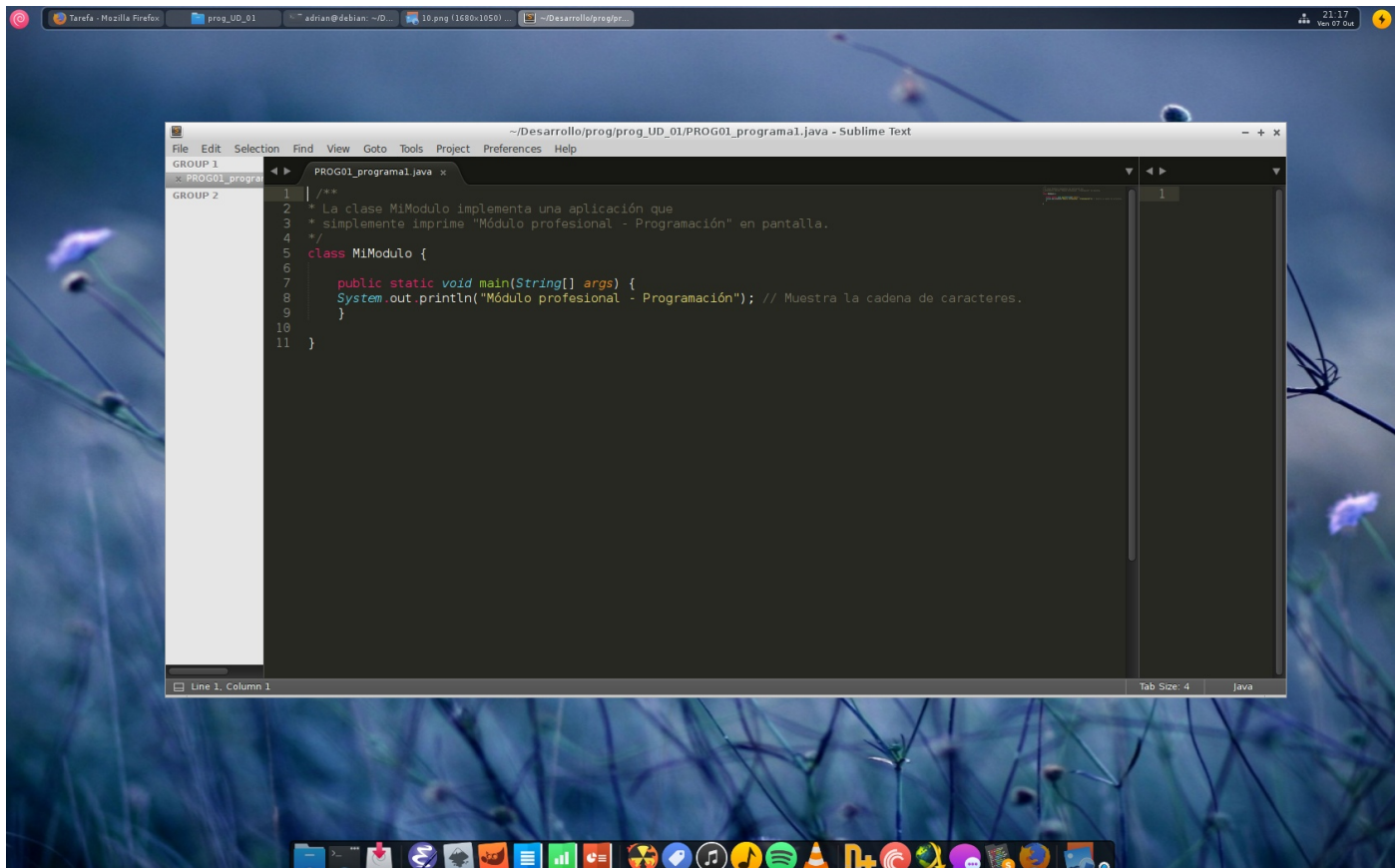
Añado la ruta editando el bashrc con nano y listo.

3. Ejecuto javac y veo que me muestra las distintas opciones y todo funciona.



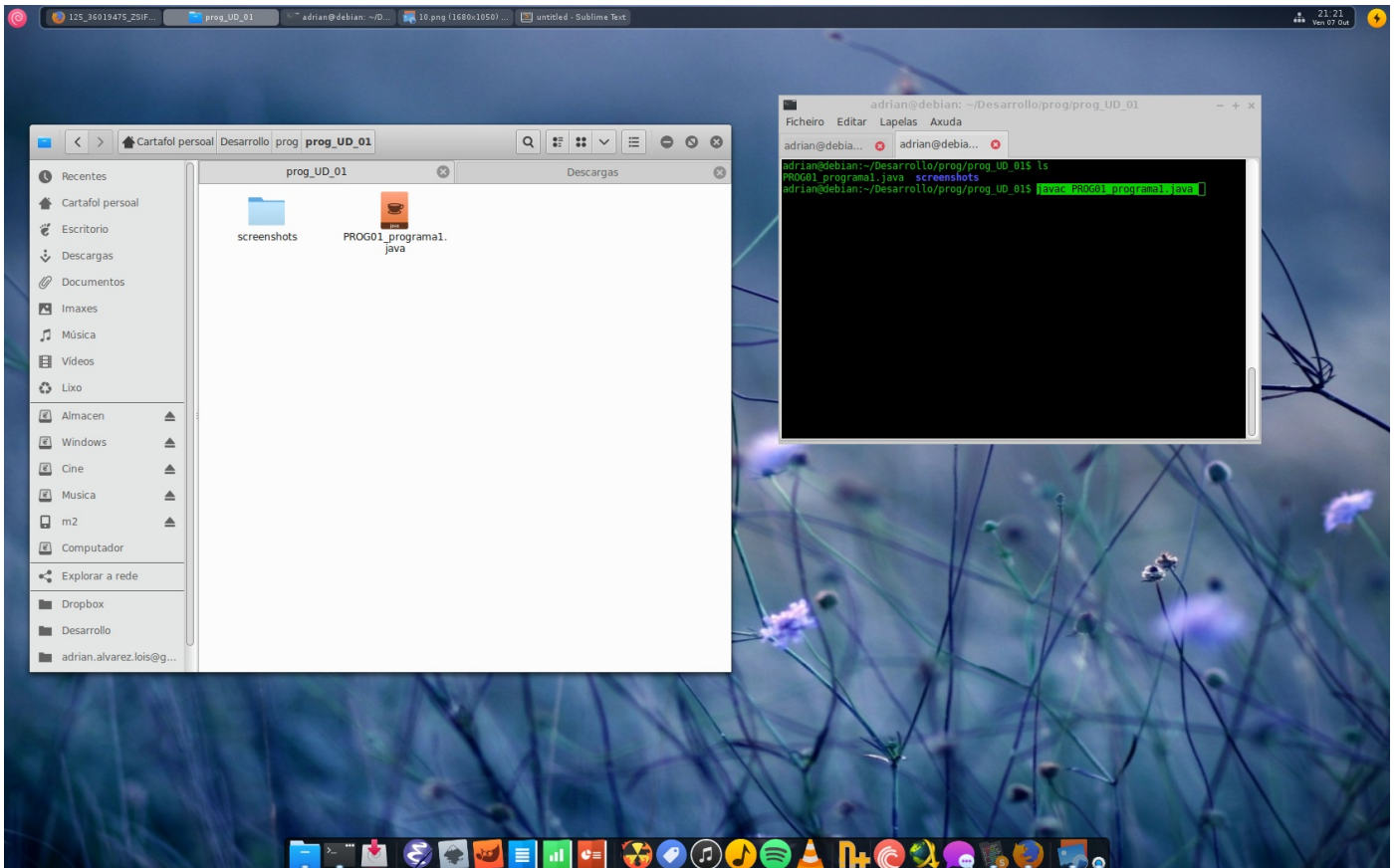
```
adrian@debian:~$ sudo nano ~/.bashrc
[sudo] password for adrian:
adrian@debian:~$ javac
Usage: javac <options> <source files>
where possible options include:
  -g               Generate all debugging info
  -g:none          Generate no debugging info
  -g:{lines,vars,source} Generate only some debugging info
  -nowarn          Generate no warnings
  -verbose         Output messages about what the compiler is doing
  -deprecation     Output source locations where deprecated APIs are used
  -classpath <path> Specify where to find user class files and annotation processors
  -cp <path>       Specify where to find user class files and annotation processors
  -sourcepath <path> Specify where to find input source files
  -bootclasspath <path> Override location of bootstrap class files
  -extdirs <dirs>   Override location of installed extensions
  -endorsedirs <dirs> Override location of endorsed standards path
  -proc:{none,only} Control whether annotation processing and/or compilation is done.
  -processor <class1[,<class2>,<class3>...] Names of the annotation processors to run; bypasses default discovery process
```

4. Tras crear una carpeta para tal fin y con mi editor de texto favorito creo el archivo *PROG01_programa1.java*

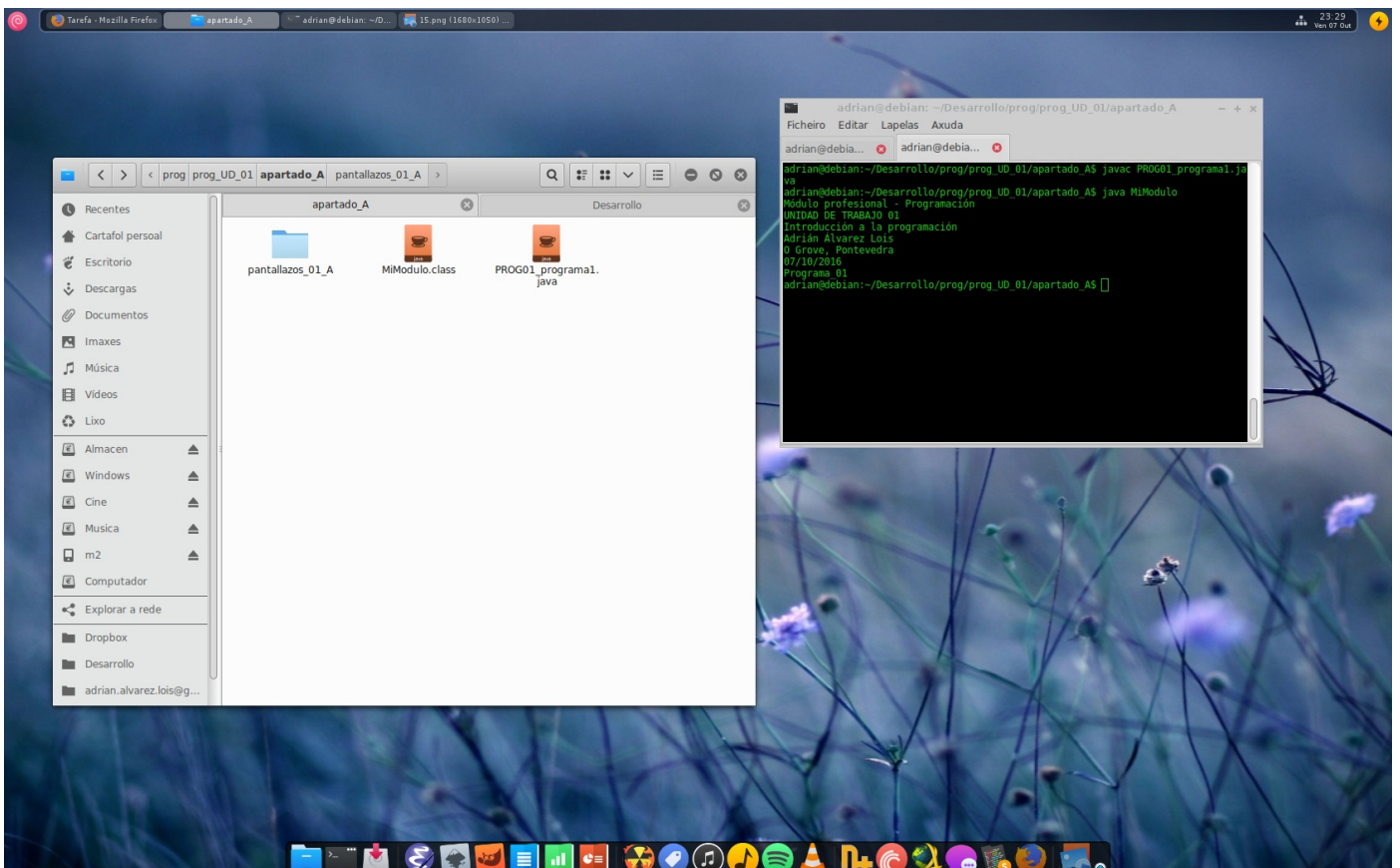


```
~/Desarrollo/prog/prog_UD_01/PROG01_programa1.java - Sublime Text
1  /**
2   * La clase MiModulo implementa una aplicación que
3   * simplemente imprime "Módulo profesional - Programación" en pantalla.
4   */
5  class MiModulo {
6
7      public static void main(String[] args) {
8          System.out.println("Módulo profesional - Programación"); // Muestra la cadena de caracteres.
9      }
10
11 }
```

5. Compilo el fichero .java y se crea MiModulo.class



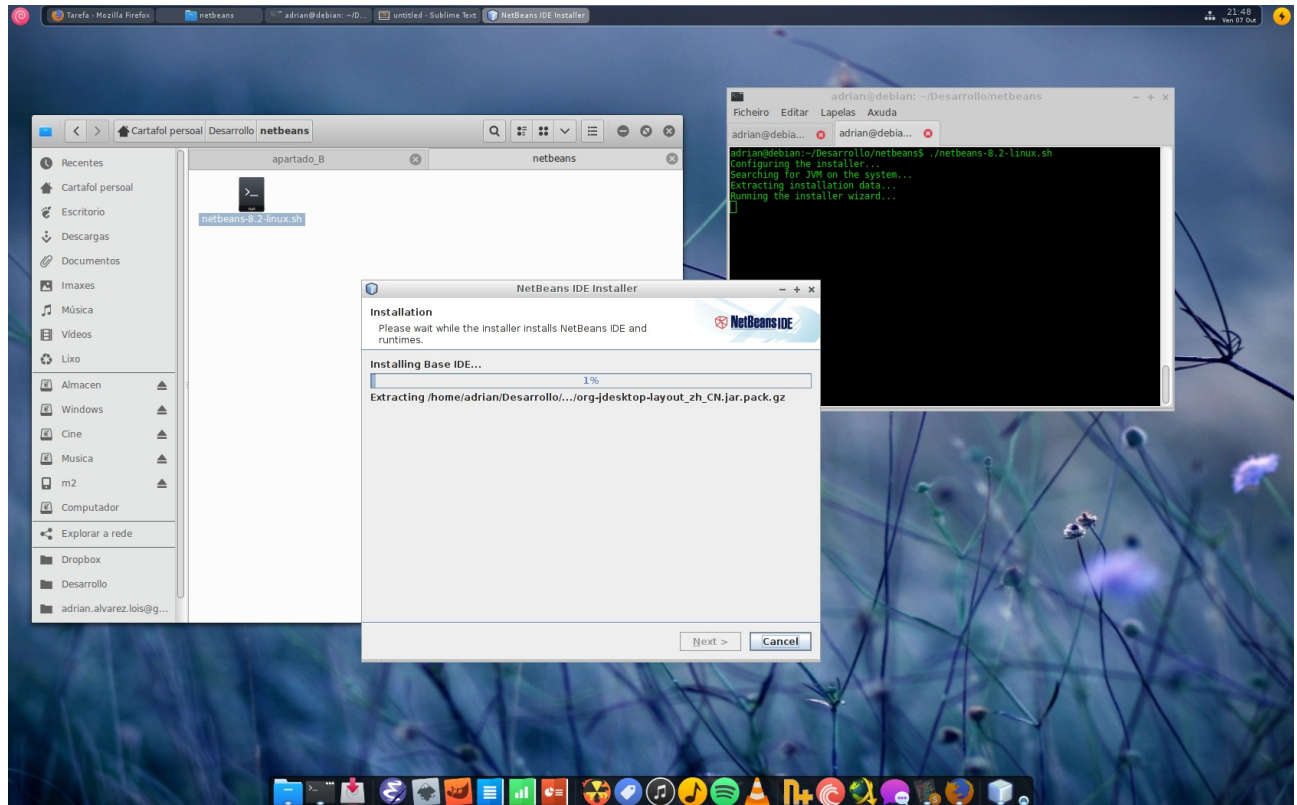
6. Ejecuto el programa y veo los resultados



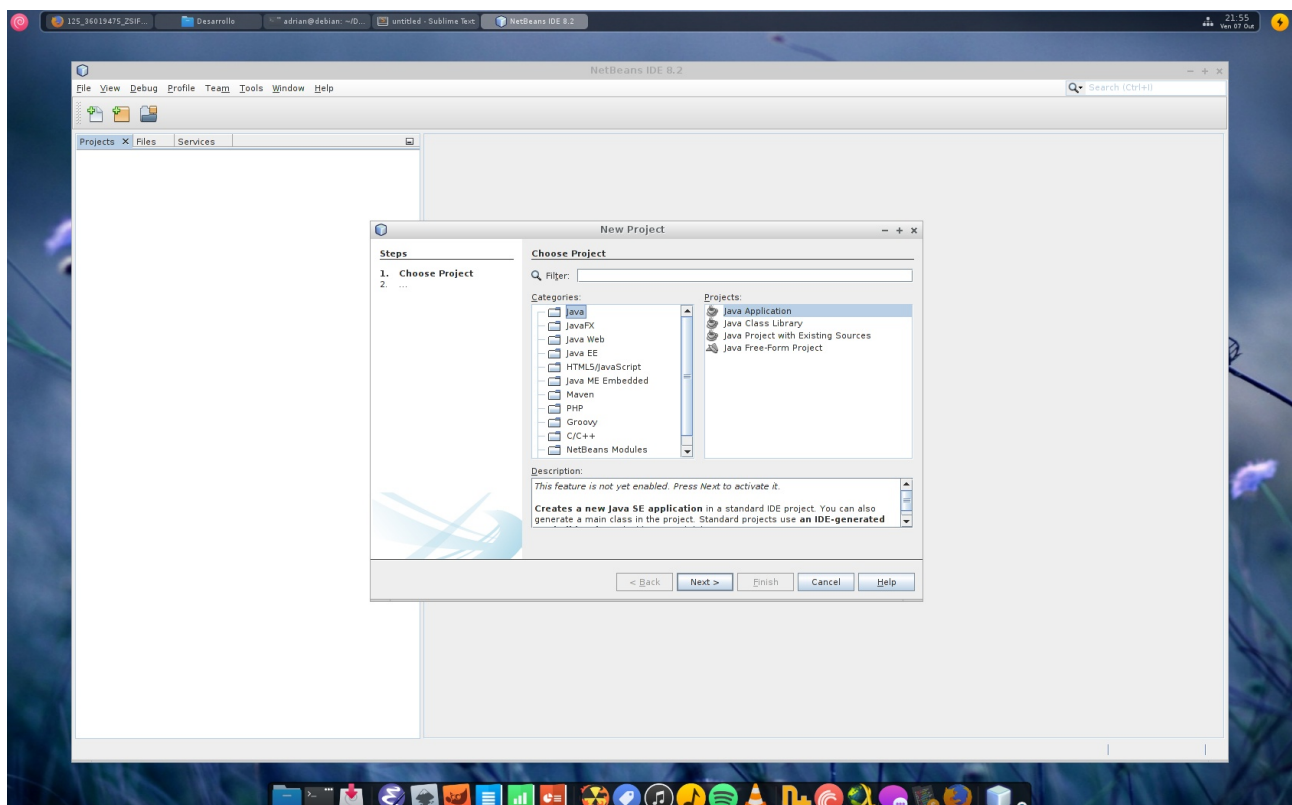
Alumno: Adrián Álvarez Lois
PROG_UD_01

Apartado B:

1. Instalo Netbeans en mi PC bajandome el .sh de la web oficial. Luego le doy permisos de ejecución con `chmod +x` y finalmente ejecuto con `./`



2. Creo mi primer proyecto en Netbeans



3. Escribo el código fuente en el archivo source y lo compilo con build. Se puede observar la ubicación de los ficheros .java y .class tal y como NetBeans los dispone al margen izquierdo. Finalmente se compila al salvar y ejecuto con Run y me muestra el resultado.

