

Seminario - parte 2

Git y Eclipse

PROGRAMACION 3

David Rizo, Felipe Sánchez, Pedro J. Ponce de León,
Francisco Moreno
Departamento de Lenguajes y Sistemas Informáticos
Universidad de Alicante



Git y Eclipse

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno



Contenidos

Git y Github

- Git y GitHub
- Crear un repositorio
- Crear un Personal access token
- Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

- Operaciones con Git
- Actualizar cambios en GitHub
- Flujo de trabajo

Eclipse

- Instalación
- Entorno
- Workspace
- Interfaz
- Creación de un proyecto

Clases

- Importación clases
- Creación de clases

Ejecución

- Depuración

Generación código

Contenidos

1 Git y Github

Git y GitHub

Crear un repositorio en GitHub

Crear un Personal access token

Importar un proyecto en Eclipse desde GitHub

2 Trabajando en Eclipse con git/GitHub

Operaciones con Git

Actualizar cambios en GitHub

Flujo de trabajo con Github y Eclipse

3 Eclipse

Instalación

Entorno

Workspace

Interfaz

Creación de un proyecto

4 Clases

Importación clases

Creación de clases

5 Ejecución

Depuración

6 Generación código

Git y Eclipse

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno



Contenidos

Git y Github

Git y GitHub

Crear un repositorio

Crear un Personal access token

Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

Operaciones con Git

Actualizar cambios en GitHub

Flujo de trabajo

Eclipse

Instalación

Entorno

Workspace

Interfaz

Creación de un proyecto

Clases

Importación clases

Creación de clases

Ejecución

Depuración

Generación código

Git

Git es un sistema de **control de versiones** que permite la **gestión** de los **cambios** que se realizan sobre un conjunto de ficheros

GitHub

GitHub (<https://github.com/>) es una plataforma para el desarrollo de aplicaciones que hace uso de Git

En Programación 3 usaremos Git y GitHub para ...

- ... alojar una copia actualizada de vuestras prácticas
- ... que el profesorado haga un seguimiento de éstas

Crear un repositorio en GitHub

Para crear un repositorio para las prácticas es necesario:

- 1 Darse de alta en GitHub con el correo de la UA (@alu.ua.es): <https://github.com/join>
- 2 Crear repositorio **privado**: <https://github.com/new>
- 3 Añadir a vuestro profesor como colaborador
- 4 Descargar la plantilla de proyecto y actualizar el repositorio
- 5 Crear un *Personal access token* para poder acceder al repositorio desde Eclipse

Esto hay que hacerlo una única vez

Crear un repositorio en GitHub

Crear repositorio privado vacío (sin fichero README)

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?
[Import a repository.](#)

Owner *

xxx99-ua ▾

Repository name *

prog3-12345678

Great repository names are short and memorable. Need inspiration? How about [shiny-palm-tree?](#)

Description (optional)



Public

Anyone on the internet can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

Add a README file

This is where you can write a long description for your project. [Learn more.](#)

Add .gitignore

Choose which files not to track from a list of templates. [Learn more.](#)

Choose a license

A license tells others what they can and can't do with your code. [Learn more.](#)

Create repository

Git y Eclipse

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno

lsi

Contenidos

Git y Github

Git y GitHub

Crear un repositorio

Crear un Personal access token

Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

Operaciones con Git

Actualizar cambios en GitHub

Flujo de trabajo

Eclipse

Instalación

Entorno

Workspace

Interfaz

Creación de un proyecto

Clases

Importación clases

Creación de clases

Ejecución

Depuración

Generación código

Seminario - parte 2.5

Crear un repositorio en GitHub

Añadir a vuestro profesor como colaborador:

Gonzalo Alcalá	jgonzalo-dlsi
Rafael Carrasco	carrasco-dlsi
Francisco Moreno	paco-dlsi

The screenshot shows the GitHub repository settings page for 'pierre-dlsi/test-import'. The page is divided into several sections:

- Settings:** A red arrow labeled '1' points to the 'Settings' tab in the top navigation bar.
- Who has access:** This section shows two access levels: 'PRIVATE REPOSITORY' (locked) and 'DIRECT ACCESS' (unlocked). A red arrow labeled '2' points to the 'Manage access' link in the left sidebar.
- Manage access:** This section displays a message: 'You haven't invited any collaborators yet'. A red arrow labeled '3' points to the 'Invite a collaborator' button.

Git y Eclipse

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno



Contenidos

Git y GitHub

Git y GitHub

Crear un repositorio

Crear un Personal access token

Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

Operaciones con Git

Actualizar cambios en GitHub

Flujo de trabajo

Eclipse

Instalación

Entorno

Workspace

Interfaz

Creación de un proyecto

Clases

Importación clases

Creación de clases

Ejecución

Depuración

Generación código

Crear un repositorio en GitHub

Descargar la plantilla de proyecto y actualizar el repositorio (1/2)

🔒 pierre-dlsi / test-import Private

👁 Watch 0 ⭐ Star 0 🍴 Fork 0

<> Code ① Issues 0 ↗ Pull requests 0 📁 Projects 0 📖 Wiki ⚙ Settings Insights ▾

Quick setup — if you've done this kind of thing before

📄 Set up in Desktop or HTTPS SSH <https://github.com/pierre-dlsi/test-import.git> 📄

We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# test-import" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/pierre-dlsi/test-import.git
git push -u origin master
```

Botón 'Import code'

...or push an existing repository from the command line

```
git remote add origin https://github.com/pierre-dlsi/test-import.git
git push -u origin master
```

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

Git y Eclipse

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno



Contenidos

Git y GitHub

Git y GitHub

Crear un repositorio

Crear un Personal access token

Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

Operaciones con Git

Actualizar cambios en GitHub

Flujo de trabajo

Eclipse

Instalación

Entorno

Workspace

Interfaz

Creación de un proyecto

Clases

Importación clases

Creación de clases

Ejecución

Depuración

Generación código

Seminario - parte 2.7

Crear un repositorio en GitHub

Descargar la plantilla de proyecto y actualizar el repositorio (2/2)

Import your project to GitHub

Import all the files, including the revision history, from another version control system.

Your old repository's clone URL

1

Learn more about the types of [supported VCS](#).

Your existing repository

 xxx99-ua/prog3-12345678

[Change repository](#)

Cancel

2

Git y Eclipse

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno



Contenidos

Git y Github

Git y GitHub

Crear un repositorio

Crear un Personal access token

Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

Operaciones con Git

Actualizar cambios en GitHub

Flujo de trabajo

Eclipse

Instalación

Entorno

Workspace

Interfaz

Creación de un proyecto

Clases

Importación clases

Creación de clases

Ejecución

Depuración

Generación código

Crear un *Personal access token* /1

Por seguridad, desde agosto de 2021, no es posible acceder a repositorios de GitHub desde Eclipse (y git) usando la contraseña, es necesario crear un *Personal access token* y utilizarlo como si fuera la contraseña:

- Desde la web de GitHub, entrar en el apartado **Settings** del usuario (pinchando sobre el usuario, después en **Settings**), y seleccionar **Developer settings**

The screenshot shows the GitHub user settings page for the account 'xxx99-ua'. The page is divided into two main sections: 'Account settings' on the left and 'Public profile' on the right. The 'Account settings' section includes a sidebar with various categories, and the 'Public profile' section contains several form fields for user information. A red box highlights the 'Developer settings' option in the sidebar.

Account settings

- Profile
- Account
- Appearance
- Account security
- Billing & plans
- Security log
- Security & analysis
- Sponsorship log
- Emails
- Notifications
- SSH and GPG keys
- Repositories
- Packages
- Organizations
- Saved replies
- Applications
- Developer settings**
- Moderation settings
- Blocked users

Public profile

Name

Your name may appear around GitHub where you contribute or are mentioned. You can remove it at any time.

Public email

Select a verified email to display

You have set your email address to private. To toggle email privacy, go to [email settings](#) and uncheck "Keep my email address private."

Bio

Tell us a little bit about yourself

You can @mention other users and organizations to link to them.

URL

Twitter username

Company

You can @mention your company's GitHub organization to link it.

Location

All of the fields on this page are optional and can be deleted at any time, and by filling them out, you're giving us consent to share this data wherever your user profile appears.

Crear un Personal access token /2

Git y Eclipse

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno



Contenidos

Git y Github

Git y GitHub

Crear un repositorio

Crear un Personal access token

Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

Operaciones con Git

Actualizar cambios en GitHub

Flujo de trabajo

Eclipse

Instalación

Entorno

Workspace

Interfaz

Creación de un proyecto

Clases

Importación clases

Creación de clases

Ejecución

Depuración

Generación código

Seminario - parte 2.10

- Seleccionar Personal access token y pinchar en Generate new token

Settings / Developer settings

GitHub Apps

OAuth Apps

Personal access tokens

Personal access tokens

Need an API token for scripts or testing? [Generate a personal access token](#) for quick access to the [GitHub API](#).

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Generate new token

© 2021 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Docs](#) [Contact GitHub](#) [Pricing](#) [API](#) [Training](#) [Blog](#) [About](#)

Crear un Personal access token /3

- 1 Rellenar el campo Note
- 2 Seleccionar un valor para el desplegable Expiration (a No expiration)
- 3 **MUY IMPORTANTE:** seleccionar el scope repo

New personal access token

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Note

git access from Eclipse

What is this token for?

Expiration *

No expiration The token will never expire!

Beep bo! Tokens that live forever are scary. Expiration dates are highly recommended!

Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes](#).

- | | |
|--|--------------------------------------|
| <input checked="" type="checkbox"/> repo | Full control of private repositories |
| <input type="checkbox"/> repo:status | Access commit status |
| <input type="checkbox"/> repo_deployment | Access deployment status |
| <input type="checkbox"/> public_repo | Access public repositories |
| <input type="checkbox"/> repo:invite | Access repository invitations |
| <input type="checkbox"/> security_events | Read and write security events |

- 4 Pinchar en el botón Generate token

- | | |
|--|--|
| <input type="checkbox"/> read:enterprise | Read enterprise profile data |
| <input type="checkbox"/> admin:ggp_key | Full control of public user GPG keys (Developer Preview) |
| <input type="checkbox"/> write:ggp_key | Write public user GPG keys |
| <input type="checkbox"/> read:ggp_key | Read public user GPG keys |

Generate token Cancel

Git y Eclipse

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno



Contenidos

Git y Github

Git y GitHub

Crear un repositorio

Crear un Personal access token

Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

Operaciones con Git

Actualizar cambios en GitHub

Flujo de trabajo

Eclipse

Instalación

Entorno

Workspace

Interfaz

Creación de un proyecto

Clases

Importación clases

Creación de clases

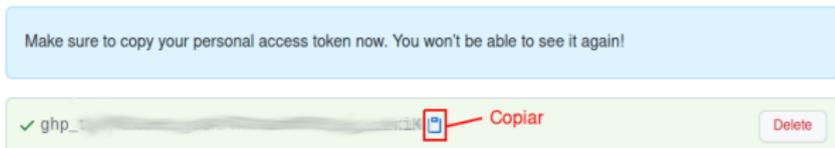
Ejecución

Depuración

Generación código

Crear un Personal access token /4

- **MUY IMPORTANTE:** guardar el token generado para usarlo más adelante como contraseña



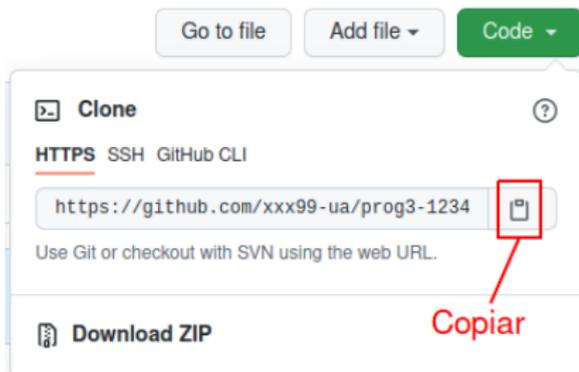
Importar un proyecto en Eclipse desde GitHub /1

Git y Eclipse

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno



- 1 Copiar la URI de vuestro repositorio desde el botón 'Code' de la página principal de vuestro repositorio de Github



Contenidos

Git y Github

Git y GitHub

Crear un repositorio

Crear un Personal access token

Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

Operaciones con Git

Actualizar cambios en GitHub

Flujo de trabajo

Eclipse

Instalación

Entorno

Workspace

Interfaz

Creación de un proyecto

Clases

Importación clases

Creación de clases

Ejecución

Depuración

Generación código

Importar un proyecto en Eclipse desde GitHub /2

- Entrar en Eclipse y seleccionar `workspace` (puedes dejar el que se sugiere)
- `File > Import`
- `Git > Projects from Git` y clic en `Next`
- Seleccionar `Clone URI` y clic en `Next`
- Introducir URI (vuestro repositorio), usuario de Github y el *Personal access token* (en el campo `Password`)

Import Projects from Git

Source Git Repository

Enter the location of the source repository.

Location

URI: Local Folder... Local Bundle File...

Host:

Repository path:

Connection

Protocol:

Port:

Authentication

User:

Password:

Store in Secure Store

Personal access token

< Back Next > Cancel Finish

Git y Eclipse

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno

lsi

Contenidos

Git y Github

Git y GitHub

Crear un repositorio

Crear un Personal access token

Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

Operaciones con Git

Actualizar cambios en GitHub

Flujo de trabajo

Eclipse

Instalación

Entorno

Workspace

Interfaz

Creación de un proyecto

Clases

Importación clases

Creación de clases

Ejecución

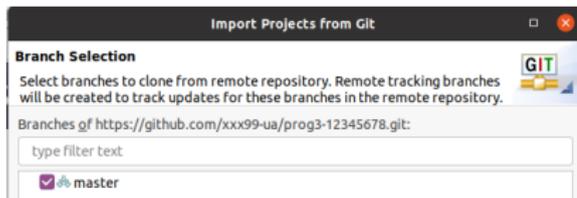
Depuración

Generación código

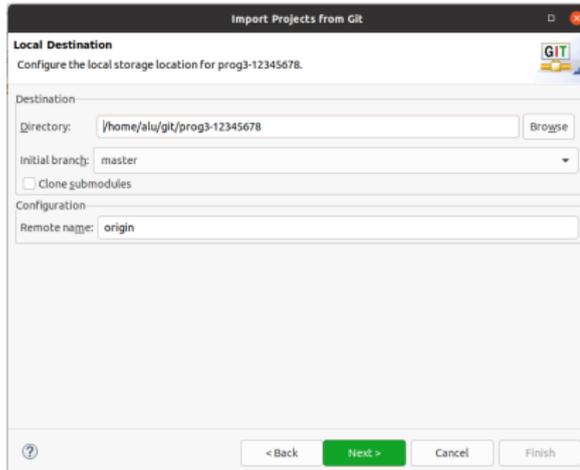
Seminario - parte 2.14

Importar un proyecto en Eclipse desde GitHub /3

7 Seleccionar `master` y clic en `Next`



8 En `Directory` especificar la ubicación donde se clonará el repositorio y clic en `Next`. Se recomienda dejar la que sugiere Eclipse, pero puedes modificarla.



Git y Eclipse

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno



Contenidos

Git y GitHub

Git y GitHub

Crear un repositorio

Crear un Personal access token

Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

Operaciones con Git

Actualizar cambios en GitHub

Flujo de trabajo

Eclipse

Instalación

Entorno

Workspace

Interfaz

Creación de un proyecto

Clases

Importación clases

Creación de clases

Ejecución

Depuración

Generación código

Seminario - parte 2.15

Importar un proyecto en Eclipse desde GitHub /4

Git y Eclipse

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno



Contenidos

Git y Github

Git y GitHub

Crear un repositorio

Crear un Personal access token

Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

Operaciones con Git

Actualizar cambios en GitHub

Flujo de trabajo

Eclipse

Instalación

Entorno

Workspace

Interfaz

Creación de un proyecto

Clases

Importación clases

Creación de clases

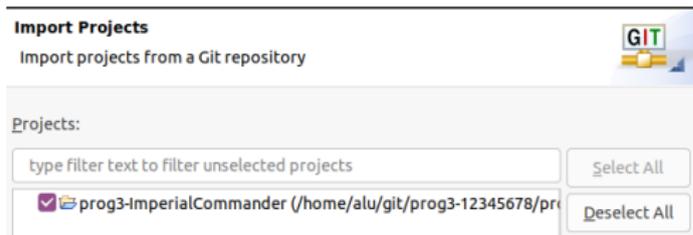
Ejecución

Depuración

Generación código

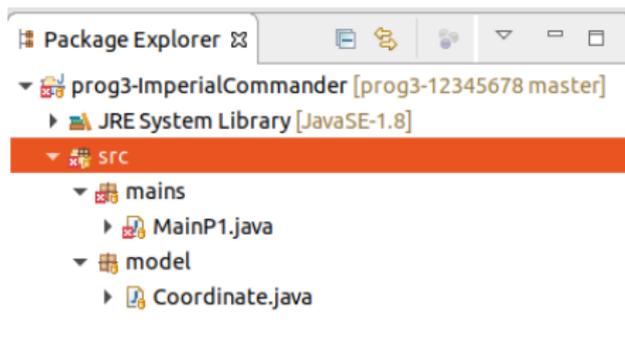
Seminario - parte 2.16

- 9 Seleccionar la opción `Import existing Eclipse projects` y pulsa `Next`
- 10 Selecciona el proyecto (`prog3-ImperialCommander`) y luego pulsa `Finish`.



Importar un proyecto en Eclipse desde GitHub /5

Debe aparecerte ahora el proyecto en la pestaña 'Package Explorer' de Eclipse:



(la marca roja en `MainP1.java` se debe a que hay errores de compilación porque la clase `Coordinate.java` no tiene código)
El proyecto está configurado para:

- Utilizar el JDK preconfigurado por Eclipse (JDK 1.8 en los laboratorios)
- Utilizar la codificación de caracteres UTF-8 al crear archivos de código fuente.
- Utilizar saltos de línea estilo Unix (carácter '`\n`' para finales de línea) en los ficheros de código fuente.

IMPORTANTE

- Debes asumir que la copia principal de tu proyecto es **siempre** la que hay en GitHub
- En cada ordenador en el que trabajes (en casa, en el laboratorio, ...):
 - 1 La primera vez, debes importar el repositorio desde GitHub (como se ha explicado antes)
 - 2 Las siguientes veces es muy importante que hagas esto:
 - Antes de empezar a trabajar, selecciona la opción `Team > Pull` para bajar a este ordenador la última copia desde GitHub
 - Después, desde la vista `Package Explorer` selecciona la carpeta del proyecto, clic en botón derecho del ratón y selecciona la opción `Refresh` (o pulsar `F5`)
 - Al acabar, antes de cerrar Eclipse, debes guardar los cambios otra vez en el repositorio de GitHub (como se explica más adelante)

Contenidos

Git y GitHub

- Git y GitHub
- Crear un repositorio
- Crear un Personal access token
- Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

Operaciones con Git

- Actualizar cambios en GitHub
- Flujo de trabajo

Eclipse

- Instalación
- Entorno
- Workspace
- Interfaz
- Creación de un proyecto

Clases

- Importación clases
- Creación de clases

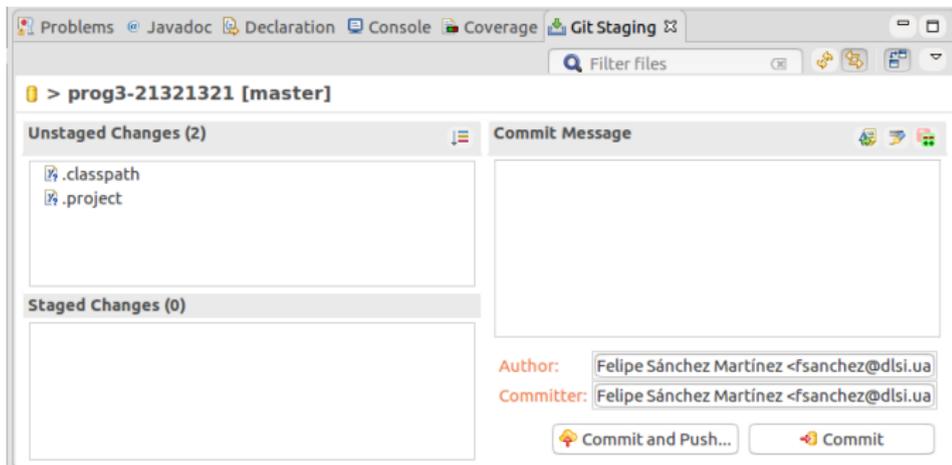
Ejecución

- Depuración

Generación código

Actualizar cambios en Github

- 1 Desde la vista `Package Explorer` selecciona la carpeta del proyecto, clic en botón derecho del ratón y selecciona la opción `Team > Commit`



- 2 Arrastrar los ficheros deseados desde `Unstaged Changes` a `Staged Changes` (si no están ya allí)
- 3 Escribir mensaje descriptivo en `Commit Message`
- 4 Clic en `Commit and push...`
- 5 Clic en `Ok`

Git y Eclipse

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno



Contenidos

Git y Github

- Git y GitHub
- Crear un repositorio
- Crear un Personal access token
- Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

- Operaciones con Git
- Actualizar cambios en GitHub

Flujo de trabajo

Eclipse

- Instalación
- Entorno
 - Workspace
 - Interfaz
- Creación de un proyecto

Clases

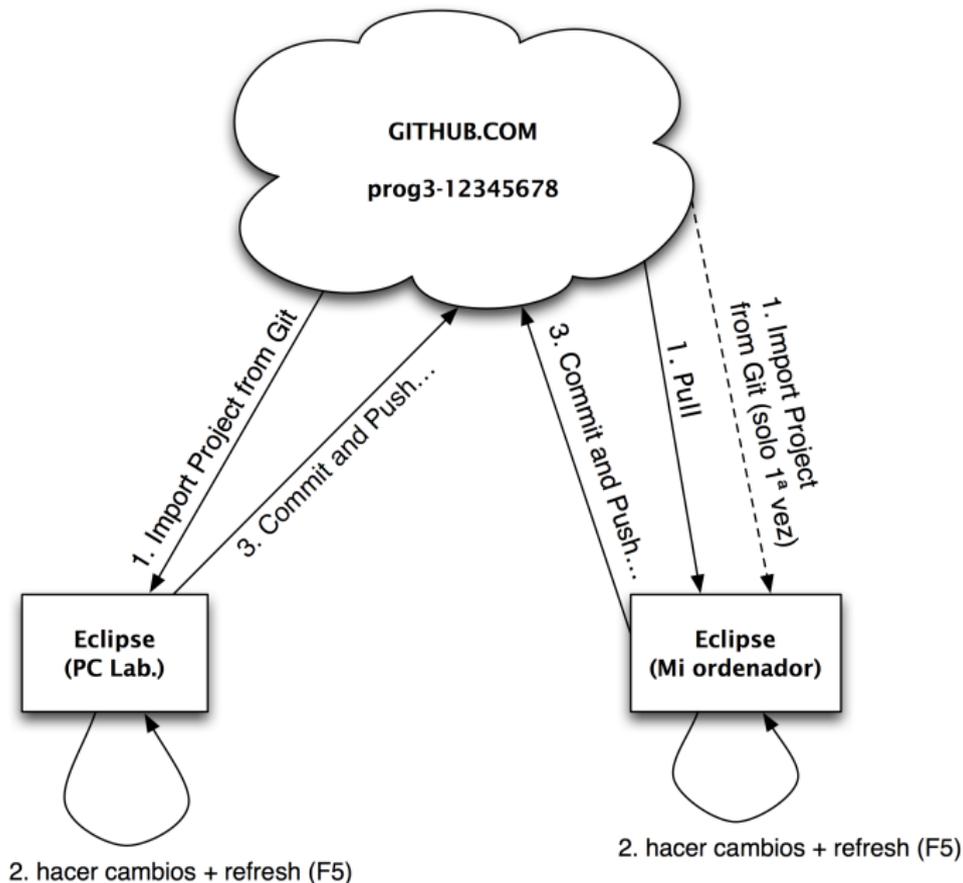
- Importación clases
- Creación de clases

Ejecución

- Depuración

Generación código

Resumen: Flujo de trabajo con Github y Eclipse



Git y Eclipse

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno



Contenidos

Git y Github

- Git y GitHub
- Crear un repositorio
- Crear un Personal access token
- Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

- Operaciones con Git
- Actualizar cambios en GitHub

Flujo de trabajo

Eclipse

- Instalación
- Entorno
- Workspace
- Interfaz
- Creación de un proyecto

Clases

- Importación clases
- Creación de clases

Ejecución

- Depuración

Generación código



En *Programación 3* usaremos la versión 2019-06, que incluye Git de serie (aunque las versiones posteriores deben funcionar bien)

- Localizado en <https://www.eclipse.org/downloads/packages/>
- Descargar *Eclipse IDE for Java Developers*
- Descomprimir y arrancar el ejecutable `eclipse`

Contenidos

Git y Github

- Git y GitHub
- Crear un repositorio
- Crear un Personal access token
- Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

- Operaciones con Git
- Actualizar cambios en GitHub
- Flujo de trabajo

Eclipse

Instalación

- Entorno
- Workspace
- Interfaz
- Creación de un proyecto

Clases

- Importación clases
- Creación de clases

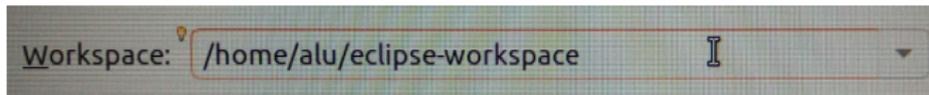
Ejecución

- Depuración

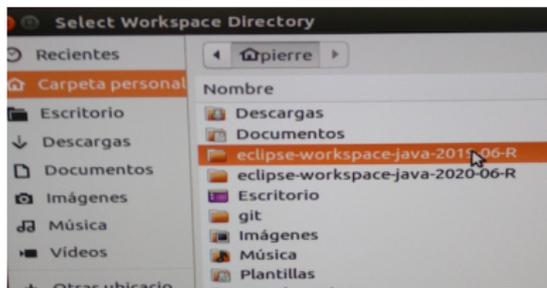
Generación código

Eclipse - Workspace

- Eclipse guarda toda su configuración en un directorio que denomina *workspace*
- Cuando iniciamos el entorno debemos decir dónde guardar el *workspace*



- Selecciona uno de estos (en el laboratorio):



- Podemos cambiar de *workspace* cuando queramos pulsando en `File>Switch workspace`

Git y Eclipse

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno



Contenidos

Git y GitHub

- Git y GitHub
- Crear un repositorio
- Crear un Personal access token
- Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

- Operaciones con Git
- Actualizar cambios en GitHub
- Flujo de trabajo

Eclipse

- Instalación
- Entorno

Workspace

- Interfaz
- Creación de un proyecto

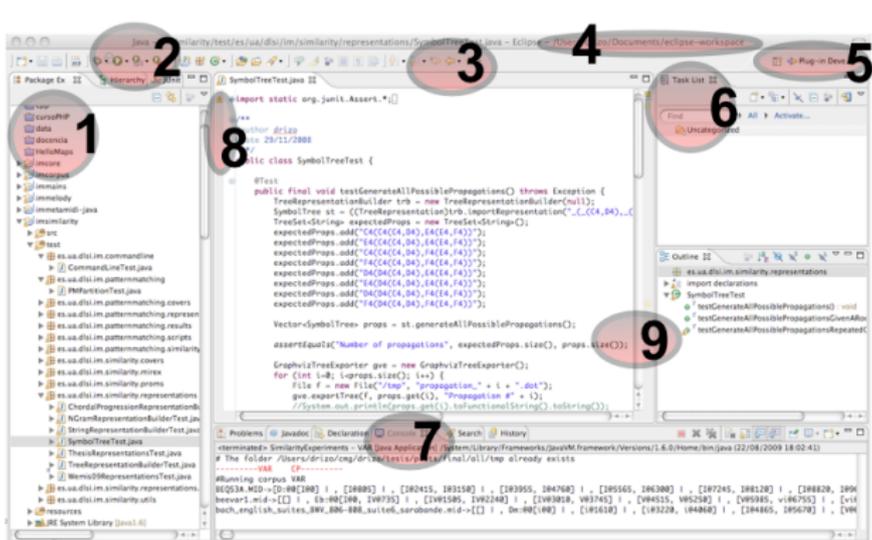
Clases

- Importación clases
- Creación de clases

Ejecución

- Depuración

Generación código



Herramientas y ayudas visuales

- 1 Proyectos y paquetes
- 2 Ejecución y depuración
- 3 Navegación por ficheros
- 4 Workspace activo
- 5 Perspectiva
- 6 Una vista: tareas
- 7 Consola
- 8 Breakpoints, enlace para solución de errores
- 9 Errores, warnings, TO-DO

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno



Contenidos

Git y Github

Git y GitHub

Crear un repositorio

Crear un Personal access token

Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

Operaciones con Git

Actualizar cambios en GitHub

Flujo de trabajo

Eclipse

Instalación

Entorno

Workspace

Interfaz

Creación de un proyecto

Clases

Importación clases

Creación de clases

Ejecución

Depuración

Generación código

Creación de proyectos

- File > New > Java project
 - Nombre del proyecto
 - *contents*: seleccionar directorio nuestro o dejar el del workspace
- Esto crea un directorio que contiene por defecto:
 - `bin, src`
 - Los ficheros ocultos `.project` y `.classpath`
 - Estos ficheros contienen los metadatos del proyecto
 - Cuando queramos llevarnos a otro ordenador un proyecto *eclipse* los usará para identificar un directorio como contenedor de un proyecto

Git y Eclipse

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno



Contenidos

Git y GitHub

- Git y GitHub
- Crear un repositorio
- Crear un Personal access token
- Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

- Operaciones con Git
- Actualizar cambios en GitHub
- Flujo de trabajo

Eclipse

- Instalación
- Entorno
- Workspace
- Interfaz

Creación de un proyecto

Clases

- Importación clases
- Creación de clases

Ejecución

- Depuración

Generación código

Importación de Clases

Podemos importar ficheros `.java` de clases escritas fuera de *eclipse* simplemente copiando los ficheros en el navegador de ficheros del sistema operativo y pegándolos en Eclipse en la vista de paquetes.

Actividad

Añadir el fichero de código fuente `Coordinate.java` de la práctica 1 al paquete `model` del proyecto que acabamos de importar de GitHub.

Git y Eclipse

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno



Contenidos

Git y Github

- Git y GitHub
- Crear un repositorio
- Crear un Personal access token
- Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

- Operaciones con Git
- Actualizar cambios en GitHub
- Flujo de trabajo

Eclipse

- Instalación
- Entorno
- Workspace
- Interfaz
- Creación de un proyecto

Clases

- Importación clases
- Creación de clases

Ejecución

- Depuración

Generación código

- Creación con `File > New > Class`
- Especificamos nombre, paquete, y opcionalmente si queremos que nos añada un `main`

Actividad

- Crear una clase denominada *Board* en el paquete `model` y añade el atributo privado `int size`. Escribiendo sobre éste `/**` y pulsando *enter* nos ayudará a crear la documentación *javadoc*.
- Crear el constructor `public Board(int size)` y añade de la misma forma la documentación.
- Si tenemos algún error usaremos las ayudas que aparecen en la barra izquierda del editor de código.



- Git y GitHub
- Crear un repositorio
- Crear un Personal access token
- Importar proyecto GitHub

- Operaciones con Git
- Actualizar cambios en GitHub
- Flujo de trabajo

- Instalación
- Entorno
- Workspace
- Interfaz
- Creación de un proyecto

- Importación clases
- Creación de clases

- Depuración

Ejecución

- Dado que un proyecto puede tener varios ficheros con un método `main` lo más sencillo para ejecutar es pulsar con el botón derecho sobre la clase que contiene el `main` a ejecutar y pulsar en `Run as > Java application`.
- Esto crea una configuración de ejecución (menú `Run > Run configurations`), donde podemos añadir parámetros adicionales a la ejecución

Actividad

En línea de comandos esto sería equivalente a:

- Abrir un terminal
- Situarse en el directorio del proyecto.
- Ejecutar `java -cp bin mains.MainPl` (*Eclipse* automáticamente compila las clases y las deja en `bin`).

Git y Eclipse

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno



Contenidos

Git y GitHub

- Git y GitHub
- Crear un repositorio
- Crear un Personal access token
- Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

- Operaciones con Git
- Actualizar cambios en GitHub
- Flujo de trabajo

Eclipse

- Instalación
- Entorno
- Workspace
- Interfaz
- Creación de un proyecto

Clases

- Importación clases
- Creación de clases

Ejecución

- Depuración

Generación código

Depuración

- Pulsando en el menú `Run > Debug` (también en la barra de herramientas) se arranca la depuración de nuestra aplicación.
- Si queremos evaluar un elemento concreto en un punto determinado debemos fijar un *breakpoint*
- Al arrancar la depuración se cambia la *perspectiva* de *Eclipse* a *Debug*.

Actividad

- 1 Pon un *breakpoint* en la primera línea de código de la función `main` en `MainP1.java` y
- 2 ejecútalo línea a línea.

Ayuda

- Step into (F5)* Ejecutar paso a paso entrando en cada método.
- Step over (F6)* Ejecutar la siguiente línea completa en un solo paso.
- Step return (F7)* Ejecutar el resto del método actual y retornar al punto de llamada.
- Resume (F8)* Continuar la ejecución hasta el siguiente breakpoint (o fin de la aplicación).
- Run to line (^R)* Continuar la ejecución hasta la línea donde está situado el cursor.

Git y Eclipse

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno



Contenidos

Git y GitHub

- Git y GitHub
- Crear un repositorio
- Crear un Personal access token
- Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

- Operaciones con Git
- Actualizar cambios en GitHub
- Flujo de trabajo

Eclipse

- Instalación
- Entorno
- Workspace
- Interfaz
- Creación de un proyecto

Clases

- Importación clases
- Creación de clases

Ejecución

Depuración

Generación código

Generación de código

- La implementación de algunas operaciones como `equals` o `toString` suele ser rutinaria
- *Eclipse* nos ayuda a realizarlo pulsando con el botón derecho en el código de la clase y seleccionando `Source` > `Generate` Esto generará un código base que luego será fácil modificar.

Actividad

Generar el método `toString()` para `Board`.

Git y Eclipse

David Rizo, Felipe Sánchez, Pedro J. Ponce de León, Francisco Moreno



Contenidos

Git y GitHub

- Git y GitHub
- Crear un repositorio
- Crear un Personal access token
- Importar proyecto GitHub

Trabajando en Eclipse con git/GitHub

- Operaciones con Git
- Actualizar cambios en GitHub
- Flujo de trabajo

Eclipse

- Instalación
- Entorno
- Workspace
- Interfaz
- Creación de un proyecto

Clases

- Importación clases
- Creación de clases

Ejecución

- Depuración

Generación código