

Diplomado en Diseño Web

Dia 1

Barragán Jiménez Jonathan

Centro ADM

23 de marzo de 2019

Índice

- 1** Setup de entorno de desarrollo local.
 - Presentación del curso
 - Manejador de Paquetes
 - Instalación

- 2** Primeros pasos en terminal terminal

- 3** Git

Índice

1 Setup de entorno de desarrollo local.

- Presentación del curso
- Manejador de Paquetes
- Instalación

2 Primeros pasos en terminal terminal

3 Git

Índice

1 Setup de entorno de desarrollo local.

- Presentación del curso
- Manejador de Paquetes
- Instalación

2 Primeros pasos en terminal terminal

3 Git

Presentación del curso

- Repositorio del curso.

https:

`//github.com/CentroADM/DiplomadoDesarrolloWeb`

Índice

1 Setup de entorno de desarrollo local.

- Presentación del curso
- Manejador de Paquetes
- Instalación

2 Primeros pasos en terminal terminal

3 Git

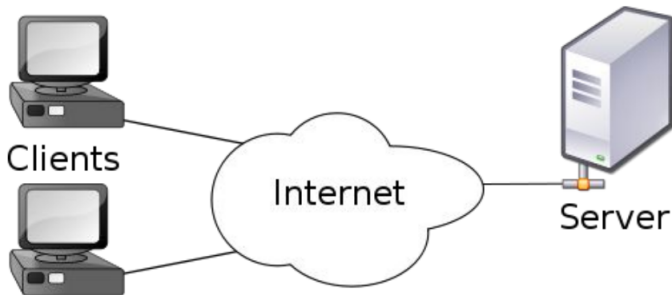
¿Qué es un manejador de paquetes?

Manejador de paquetes

Un manejador (gestor) de paquetes mantiene un registro del software que está instalado en su computadora, y le permite instalar software nuevo, actualizarlo a versiones más recientes, o eliminar software de una manera sencilla.

Aplicaciones WEB

¿Cómo funcionan?



Índice

1 Setup de entorno de desarrollo local.

- Presentación del curso
- Manejador de Paquetes
- Instalación

2 Primeros pasos en terminal terminal

3 Git

Instalación: Manejador de Paquetes

Manejador de paquetes

- Windows

`https:`

`//www.microsoft.com/web/downloads/platform.aspx`

- macOS

`https://brew.sh/index_es`

- Linux Ubuntu/Debian `apt-get`

Instalación:git

git

- Windows
<https://git-scm.com/>
- macOS **brew install git**
- Linux Ubuntu/Debian **sudo apt-get install git**

Instalación:Mysql

git

- Windows
<https://dev.mysql.com/downloads/installer/>
- macOS **brew install mysql**
- Linux Ubuntu/Debian **sudo apt-get install mysql**

Índice

1 Setup de entorno de desarrollo local.

- Presentación del curso
- Manejador de Paquetes
- Instalación

2 Primeros pasos en terminal terminal

3 Git

Terminal (línea de comandos)

¿Qué es una terminal?

Un terminal es una forma de acceder al sistema sin utilizar la interfaz gráfica, es decir, realizar todo tipo de tareas en formato texto. La forma de utilizar el sistema de este modo es mediante órdenes (comandos).

Terminal (línea de comandos)

Terminal

- Windows **window** > **cmd**
- macOS **Finder** > **Aplicaciones** > **Utilidades**
- Linux **ctrl + alt + t**

Terminal: Comandos basicos

Terminal

Existen muchos comandos utiles para realizar ciertas opreaciones de manera rapida y eficiente. A continuacion se listan algunos comandos basicos para aprender la utilidad de usar terminal.^a

^aSe puede intalar una terminal en windows donde los comandos en linux y en macOS sean validos

Terminal: Comandos basicos

macOS/Linux

- **LS** permite listar el contenido de un directorio.
- **CD** permite cambiar de directorio.
- **TOUCH** Crea un archivo vacio.
- **CP** Copia un archivo o directorio origen a un archivo o directorio destino.
- **MKDIR** Crea un directorio (carpeta) nueva tomado encuentra la ubicacion actual.

Windows

- **DIR** permite listar el contenido de un directorio.
- **CD** permite cambiar de directorio.
- **CLS** Limpia nuestra terminal dejandola como recien abierta.
- **EXIT** Cierra la ventana de la consola.

Terminal: Comandos basicos

macOS/Linux

- **MV** Mueve un archivo a una ruta especifica.
- **RM** Borra un archivo o un directorio.
- **CLEAR** Limpia nuestra terminal dejandola como recien abierta.
- **EXIT** Cierra la ventana de la consola.

Windows

- **COPY** Copia una omas archivos en el directorio destino.
- **MOVE** Mueve un archivo a una ruta especifica.
- **DEL** Borra un archivo o un directorio.

Índice

1 Setup de entorno de desarrollo local.

- Presentación del curso
- Manejador de Paquetes
- Instalación

2 Primeros pasos en terminal terminal

3 Git

Git

Git

Es un software de control de versiones diseñado por Linus Torvalds, pensado en la eficiencia y la confiabilidad del mantenimiento de versiones.

Características importantes

- Fomenta el desarrollo no lineal a través de la gestión de ramas.
- El manejo del código se puede desarrollar de forma distribuida al general copias locales del historial del desarrollo de un proyecto.
- Los cambios de un repositorio remoto se importan como ramas adicionales y pueden ser fusionados de la misma manera que una rama local.

Comandos basicos de GIT

Comandos básicos de GIT	
<code>sudo apt-get update</code> <code>sudo apt-get install git</code>	Instalación de GIT
<code>git init</code>	Inicializar un nuevo repositorio
<code>git status</code>	Checa el estatus del repositorio actual
<code>git add nombre_archivo</code> <code>git add.</code>	<p>Con el nombre del archivo se registra dicho cambio en el caso de alguna creación, con el punto (.) se preparan todos los cambios realizados en el proyecto actual.</p> <p>Lista de cambios preparados para subir en el siguiente commit.</p>
<code>git commit</code> <code>git commit -m 'initial commit of my project'</code>	Genera un archivo de los cambios realizados en el proyecto, en el archivo que se mostrará se agrega una descripción del cambio realizado.

Comandos basicos de GIT

<code>git add . && git commit -m "Cambionumeros"</code>	Add y commit en una sola línea
<code>git log</code>	Muestra la lista de cambios hechos (commits)
<code>git log --oneline --decorate --all --graph --since=2018-12-04</code>	
<code>git diff</code>	Muestra las diferencias de los cambios hechos y que no han sido añadidos a un commit
<code>git diff HEAD~1 HEAD</code>	
<code>git checkout<file></code>	
<code>git branch -a</code>	Muestra la lista de las ramas creadas
<code>git branch -d branch_name</code>	Borrar un branch
<code>git branch branch_name</code>	Creación de una nueva rama basada en la rama actual
<code>git checkout -b hotFix</code>	Creación de una nueva rama y ubicación dentro de esa rama

Comandos basicos de GIT

<code>git merge branch_name</code>	<p>Para este paso suponemos un ejemplo Tenemos el branch develop y feature, y queremos integrar la rama feature a develop por lo que debemos hacer los siguientes pasos:</p> <ol style="list-style-type: none">1. Ubicarse en la rama develop mediante el comando: <code>git checkout develop</code>2. Ingresar el comando <code>git merge feature</code>
<code>git revert -m 1 SHA1_merge</code>	Revierte un merge, especificando el SHA1 del merge que se quiere revertir
<code>git revert HEAD</code> <code>git revert --no-commit HEAD</code> <code>git revert --no-commit HEAD~1</code> <code>git revert --continue</code>	Revierte un commit hecho sobre la rama actual Revertir dos o más commit junto al mismo comando de revert para ser tomados en cuenta como uno solo

Comandos basicos de GIT

```
git merge branch_name
```

Para este paso suponemos un ejemplo
Tenemos el branch develop y feature, y
queremos integrar la rama feature a develop
por lo que debemos hacer los siguientes pasos:

1. Ubicarse en la rama develop mediante
el comando:

git checkout develop

2. Ingresar el comando

git merge feature

```
git revert HEAD
```

Revierte un commit hecho sobre la rama actual