

Hector Luis Guerra Pichardo
2019-8533



Nombres y apellidos:

Hector Luis Guerra Pichardo

Matricula:

2019-8533

Materia:

Programación para mecatrónicos

Profesor:

Carlos Pichardo

NAME	CLASS	SPEAKER	DATE & TIME																																
Hector L. Guerra P.	Programación	Carlos Pichardo																																	
Title: <u>Comandos del Git</u>																																			
Keyword VCS = Version Control Systems	Topic Copias instantáneas, no diferencias. La principal diferencia entre Git y cualquier otro VCS es que los otros VCS almacenan sus datos como una lista de cambios en los archivos, mientras que, Git no maneja ni almacena sus datos como un conjunto de copias instantáneas de un sistema de un sistema de archivos.																																		
Questions	Esta es una diferencia importante entre Git y prácticamente todos los demás VCS. Haciendo que Git reconsidere casi todos los aspectos del control de versiones que muchos de los demás sistemas copiaron de la generación. <table border="1"> <thead> <tr> <th>Version 1</th> <th>V2</th> <th>V3</th> <th>V4</th> </tr> </thead> <tbody> <tr> <td>File A</td> <td>A1</td> <td>→</td> <td>A2</td> </tr> <tr> <td>File B</td> <td>—</td> <td>→</td> <td>A2</td> </tr> <tr> <td>File C</td> <td>A1</td> <td>→</td> <td>A2</td> </tr> </tbody> </table> <p>Almacenamiento de datos como cambios en versión de la base de archivos</p> <table border="1"> <thead> <tr> <th>Version 1</th> <th>V2</th> <th>V3</th> <th>V4</th> </tr> </thead> <tbody> <tr> <td>File A</td> <td>A1</td> <td>A2</td> <td>A3</td> </tr> <tr> <td>File B</td> <td>B1</td> <td>B2</td> <td>B3</td> </tr> <tr> <td>File C</td> <td>C1</td> <td>C2</td> <td>C3</td> </tr> </tbody> </table> <p>Almacenamiento de datos como instantáneas del proyecto a través del tiempo</p>			Version 1	V2	V3	V4	File A	A1	→	A2	File B	—	→	A2	File C	A1	→	A2	Version 1	V2	V3	V4	File A	A1	A2	A3	File B	B1	B2	B3	File C	C1	C2	C3
Version 1	V2	V3	V4																																
File A	A1	→	A2																																
File B	—	→	A2																																
File C	A1	→	A2																																
Version 1	V2	V3	V4																																
File A	A1	A2	A3																																
File B	B1	B2	B3																																
File C	C1	C2	C3																																
Summary:																																			

NAME	CLASS	SPEAKER	DATE & TIME
Hector L Guerra P.	Programación	Carlos Pichardo	
Title <u>Comandos del Git</u>			
Keyword	Topic Configurando Git por primera vez Git trae una herramienta llamada <u>git config</u> que te permite obtener y establecer variables de configuración que controlan el aspecto		
Questions	<p>1. Archivo <u>etc/gitconfig</u>: Contiene valores para todos los usuarios del sistema y todos los repositorios. Si pasas la opción <u>--system</u> a <u>git config</u>, lee y escribe específicamente en ese archivo.</p> <p>2. Archivo <u>~/.gitconfig</u> o <u>~/.git/config</u>: este archivo es específico a tu usuario. Puede hacer que Git lea y escriba específicamente en este archivo pasando la opción <u>--global</u>.</p> <p>3. Archivo <u>config</u> en el directorio de Git (es decir <u>.git/config</u>) del repositorio que estás utilizando actualmente. Este archivo es específico al repositorio actual.</p>		
Summary:			

NAME	CLASS	SPEAKER	DATE & TIME
Hector L Guerra P.	Programación	Carlos Pichardo	
Title <u>Comandos del Git</u>			
Keyword	Topic <u>Tu identidad</u> Lo primero que debes hacer cuando instalas Git es establecer tu nombre de usuario y dirección de correo electrónico. Esto es importante porque los "commits" de Git usan esta información, y es introducida de manera inmutable en los commits que envías. \$ git config --global user.name "John Doe" \$ git config --global user.email johndoe@example.com		
Questions	Tu editor Ahora que tu identidad está configurada, puedes elegir el editor de texto por defecto que se utilizará cuando Git necesite que introduzcas un mensaje. Si no indicas nada, Git usa el editor por defecto de tu sistema, que generalmente es Vim. Si quieres usar otro editor de texto como Emacs, puedes hacer lo siguiente \$ git config --global core.editor emacs		
Summary:			

NAME	CLASS	SPEAKER	DATE & TIME
Hector L. Guerra P.	Programación	Carlos Pichardo	
Title <u>Comandos del Git</u>			
Keyword	Topic <u>Comprobando tu Configuración</u>		
Questions	<p>Si quieres comprobar tu configuración, puedes usar el comando <code>git config --list</code> para mostrar todas las propiedades que Git ha configurado:</p> <pre>\$ git config --list user.name = John Doe user.email = johndoe@example.com color.status = auto color.branch = auto color.interactive = auto color.diff = auto ...</pre>		
	<p>Puede que veas claves repetidas, porque Git usa la misma clave de distintos archivos (<code>\$cat/.gitconfig</code> y <code>~/.gitconfig</code>, por ejemplo). En ese caso, Git usa el último valor para cada clave única que ve. También puedes comprobar qué valor que Git utilizará para una clave específica ejecutando <code>git config <key></code>:</p>		
Summary:	<pre>\$ git config user.name John Doe</pre>		