

UVM

Investiga y contesta:

1. ¿Cómo se establece el nombre de usuario en git?

Puedes escoger tu nombre de usuario al crear una cuenta.

2. ¿Cómo se establece el correo del usuario en git?

El correo es ingresado al registrar tu cuenta de GitHub, sin embargo, GitHub también te provee con un correo electrónico interno. Puedes escoger cual usar en la configuración de la cuenta.

3. ¿Cómo se activa el coloreado de salida en git?

Esto se puede configurar con el comando `color.status.<slot>`

4. ¿Cómo se muestra el estado original en los conflictos de git?

Al utilizar el comando `git checkout --ours` o `--theirs`, podemos ver la version original de ambos lados del conflicto para resolverlo.

5. ¿Cómo se muestra la configuración?

El comando `git config --list` nos permite revisar la configuración del repo.

6. ¿Cómo se crea un repositorio en GIT?

Se puede crear un repositorio con ayuda del wizard guiado en la pagina principal de GitHub, o con comandos.

7. ¿Cómo se muestra el estado de un repositorio en git?

Con el comando `git.status`

8. ¿Cómo se muestra las diferencias entre versiones en git?

Para revisar las diferencias entre dos commits especificos, podemos utilizar `git diff <commit1> <commit2>`

9. ¿Cómo se puede volver a un directorio de trabajo o versión anterior?

Para restaurar el directorio del repositorio a una version anterior, podemos especificarla en el comando `git checkout <VersionDelCommit>`

10. ¿A que se refiere con un pull en git?

Un pull se refiere a “jalar” información de un repositorio y descargarla hacia nuestra computadora para trabajar en ella localmente.

11. ¿A que se refiere con un push en git?

El “push” empuja o aplica los cambios realizados en nuestra version local a el repositorio oficial.

Practica:

- Crea un repositorio local en Git
 - Comparte ese repositorio con cada uno de tus compañeros de equipo del directorio remoto.
 - Que cada compañero tenga su escritorio remoto y añada los cambios necesarios.
 - Añadan los cambios al stage.
 - Registren los cambios.
 - Suban los cambios al remoto.
 - Realiza las siguientes operaciones sobre la rama:
 - Crea su rama.
 - Ver en que rama se encuentra.
 - Cambiar de rama.
 - Ver los cambios entre ramas.
 - Fusionar ramas
 - Eliminar rama.
 - Cada compañero deberá de mostrar su rama y su usuario además de que comparte el mismo repositorio.
 - Para un pull request:
 - Crear la rama.
 - Añada su nombre de usuario en README.md en el apartado Colaboradores.
 - Sube los cambios en local.
 - Subir los cambios.
 - Ir a github y solicitar una pull request.
 - Agregar un comentario.
-

Evidencia GitHub


The screenshot shows the GitHub interface for the repository 'ModMetDS-Cumbres'. The repository is private and has 1 branch (main) and 0 tags. It has 1 commit and 4 commits in total. The commit history shows four commits: 'Eliminar.png', 'Historias de Usuario 13.pdf', 'Limpiar.png', and 'readme.txt'. The 'readme.txt' file is selected, showing its content: 'Hola! Esto esta funcionando?' and 'En este Git se cargará información relevante a nuestro proyecto de modelado y desarrollo de software.' The right sidebar shows the repository's metadata: no description, website, or topics provided; 0 stars; 1 watching; 0 forks; and no releases or packages published.

The screenshot shows the Git GUI application interface. The current repository is 'ModMetDS-Cumbres' and the current branch is 'main'. The 'Changes' tab is selected, showing 1 changed file: 'readme.txt'. The commit history for 'readme.txt' is displayed, showing three commits. The commit message for the selected commit is: 'En este Git se cargará información relevante a nuestro proyecto de modelado y desarrollo de software.' The commit message is highlighted in red. The bottom of the window shows the 'Commit to main' button.

Se requiere que se genere lo siguiente:
Codificación de botón de eliminar y botón limpiar.

Botón eliminar:

Debe de buscar en base al código y si no existe mencionar que así es.

 **Datos Inc.**

InicioBuscarActualizarEliminarLimpiar


Cuenta →

Eliminar Producto

Ingresa el código del producto a eliminar:

Search

Producto encontrado:



ERROR!
El producto que solicitaste
no se encuentra en la
base de datos. Por favor,
revisa el código introducido.


© Datos Inc. Todos los derechos reservados, 2024

Terms & Conditions

Esta interfaz será muy sencilla. Como la única tarea necesaria es eliminar un producto registrado, basta con preguntar al usuario por el código y verificar si se encuentra registrado en la base de datos. Si es así, se elimina la entrada en la tabla. En caso de que el código no se haya encontrado, se avisará al usuario con un mensaje de error.

Botón Limpiar:

Debe de limpiar todos los campos de texto, esto lo realiza llamando una función que se debe de generar en el código, es decir no escribimos el código en el botón si no que generamos una función y se manda llamar.

 **Datos Inc.**

[Inicio](#) [Buscar](#) [Actualizar](#) [Eliminar](#) [Limpiar](#)

[Cuenta →](#)



Exito!

El codigo que has ingresado se ha eliminado correctamente.



© Datos Inc. Todos los derechos reservados, 2024

[Terms & Conditions](#)

El menú Limpiar, al igual que el menú Eliminar, puede contar con una interfaz sencilla gracias a la simplicidad de la tarea a realizar. El usuario ingresa el código que desea limpiar de la base de datos, y la aplicación llama una función que lo busca y elimina el contenido de sus campos en la tabla. Es importante notar la distinción entre Limpiar y Eliminar: Eliminar borra por completo la entrada del producto, mientras que Limpiar solo deja vacios los campos pero no elimina el código del producto de la lista.

Historia de Usuario	
Número: 1	Usuario: Usuarios de la Base de Datos
Nombre historia: Implementación del boton ELIMINAR	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Puntos estimados: 4	Iteración asignada: 1
Programador responsable: Gabriel Ezeta	
Descripción: Se debe crear un boton ELIMINAR que reciba un código de producto y elimine su lugar en la tabla. Se implementará de esta manera: -Una ventana nueva que tenga como unica opcion de entrada un campo para ingresar el código del producto, y un boton de aceptar. -Al ingresar y confirmar el código, se llamará a una función que buscara el código en la tabla y eliminará la columna asociada. Si este código no existe en la tabla, mostrará un mensaje de error.	
Observaciones: Esta tabla será usada en conjunto con una interfaz de usuario que permita manipular los datos de la tabla NOMBRE.	

Historia de Usuario	
Número: 2	Usuario: Empleados con acceso a la terminal
Nombre historia: Desarrollo de interfaz de usuario, conexión con BD	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 4	Iteración asignada: 1
Programador responsable: Gabriel Ezeta	
<p>Descripción: Esta aplicación deberá ser codificada para conectarse con la tabla NOMBRE almacenada en la base de datos NOMBRE. Al conectarse, va a interactuar con ella para recuperar y almacenar registros de manera correcta. Esta interfaz contendrá los campos equivalentes a las columnas de la tabla NOMBRE, y permitira al usuario realizar cuatro tipos de operación: GUARDAR, ACTUALIZAR, ELIMINAR y LIMPIAR. Se deberá validar que la información en los campos sea correcta. Se deberá verificar que los campos fueron agregados a la tabla NOMBRE por la aplicación Se validará el funcionamiento de la interfaz con sus botones.</p>	
Observaciones: Esta es la plantilla de la ventana deseada.	

BD:

Deberás mostrar el antes y el después.

- Abre tu plantilla y genera las actividades que consideres necesarias para los pasos anteriores.
- Genera las historias de usuario.
- Pasamos a codificar.

A entregar:

- Planificación e historias de usuario en repositorio que compartirás con el docente.
 - Captura de pantalla de cada uno de los pasos solicitados.
 - BD
 - Interfaz funcionando.
 - Liga o enlace a tu repositorio.
-