



Practica individual

Nombre

Jorgenis Belén

Matricula

2022-0445

Carrera:

Desarrollo de Software

Materia:

Programación 3

Maestro:

Kelyn Tejada Belliard

Tema:

Herramientas de administración de fuentes

1- Desarrolla el siguiente Cuestionario

1-Que es Git?

Git es un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia, la confiabilidad y compatibilidad del mantenimiento de versiones de aplicaciones cuando estas tienen un gran número de archivos de código fuente. Su propósito es llevar registro de los cambios en archivos de computadora incluyendo coordinar el trabajo que varias personas realizan sobre archivos compartidos en un repositorio de código.

2-Para que funciona el comando Git init?

Esto crea un subdirectorio nuevo llamado .git, el cual contiene todos los archivos necesarios del repositorio – un esqueleto de un repositorio de Git. Todavía no hay nada en tu proyecto que esté bajo seguimiento.

4-Que es una rama?

Las ramas son parte del proceso de desarrollo diario. Las ramas de Git son un puntero eficaz para las instantáneas de tus cambios. Cuando quieres añadir una nueva función o solucionar un error, independientemente de su tamaño, generas una nueva rama para alojar estos cambios.

3-Como saber es que rama estoy?

Para saber qué ramas están disponibles y cuál es el nombre de la rama actual, ejecuta git branch .

5-Quien creo git?

Linus Torvalds en 2005

6-Cuales son los comandos más esenciales de Git?

1.
git init: Inicializa un nuevo repositorio de Git en un directorio vacío o existente.
2. **git clone [url]:** Clona un repositorio existente de Git desde una URL remota a tu máquina local.
3. **git add [archivo]:** Agrega los cambios realizados en un archivo específico al área de preparación (staging area) para que estén listos para ser confirmados.
4. **git add .:** Agrega todos los cambios realizados en los archivos modificados al área de preparación.
5. **git commit -m "mensaje":** Confirma los cambios agregados al área de preparación y crea una nueva instantánea (commit) con un mensaje descriptivo.
6. **git status:** Muestra el estado actual del repositorio, incluyendo los cambios pendientes, archivos sin seguimiento y la rama actual.
7. **git log:** Muestra el historial de commits, mostrando los commits más recientes primero.
8. **git pull:** Obtiene los cambios más recientes desde el repositorio remoto y los fusiona con tu rama actual.
9. **git push:** Envía tus commits locales al repositorio remoto.
10. **git branch:** Muestra una lista de ramas presentes en el repositorio y destaca la rama en la que te encuentras.
11. **git checkout [rama]:** Cambia a una rama específica.
12. **git merge [rama]:** Fusiona los cambios de una rama en la rama actual.
13. **git remote -v:** Muestra la lista de repositorios remotos vinculados.
14. **git rm [archivo]:** Elimina un archivo del directorio de trabajo y lo marca para que sea eliminado del repositorio en el próximo commit.
15. **git reset [archivo]:** Retira los cambios realizados en un archivo específico del área de preparación, pero conserva los cambios en el directorio de trabajo.

7-Que es git Flow?

Es un modelo alternativo de creación de ramas en Git en el que se utilizan ramas de función y varias ramas principales

8-Que es trunk based development ?

Es una estrategia de desarrollo de software que se centra en mantener una única rama principal (trunk) como la principal línea de desarrollo para todo el equipo. En lugar de crear ramas separadas para cada característica o tarea, en el Trunk Based Development, los desarrolladores trabajan directamente en la rama principal.