Se creo una cuenta en GitHub, luego se vincularon todos los programas. También se hizo una carpeta con mi Nombre y Apellido, también carpetas para cada semana y día. Hice un repositorio con el nombre de “Practicas”. Luego agregue el repositorio remoto y verifique que se agregara correctamente. De ultimo verifique el estado del repositorio y también el historial de commits. Para poder hacer este procedimiento utilice los siguientes comandos:

* cd NombreCarpeta = Entrar a una carpeta
* mkdir = Crea una carpeta
* cd .. = Salir de una carpeta
* git add = Para añadir todos los archivos nuevos
* git init = Para crear un nuevo repositorio
* git status = Enseña el estado actual del directorio
* git commit = Es como un mensaje descriptivo
* dir = Me muestra todas las carpetas en mi carpeta actual

Preguntas:

1. ¿Por qué usar un control de versiones?

A mi punto de vista es muy importante ya que nos sirve para llevar un registro de los cambios que se hacen en un proyecto así se puede ver que cambio, quien lo hizo y cuando.

2. ¿Por qué se debe usar Git Pull?

Se debe usar git pull para traer los cambios realizados en un repositorio remoto a tu copia local, si alguien más hizo cambios y subió al repositorio tu versión local no tendrá esos cambios hasta que uses git pull.

3. ¿Qué alternativas existen a GitHub?

Algunas alternativas a GitHub son GitLab, Bitbucket, SourceForge, AWS CodeCommit y GitKraken, cada una con características que se ajustan a diferentes necesidades de colaboración y almacenamiento de código

Commit

Commit es como el titulo de un archivo, descripción. El commit no se actualiza por si solo, con el push se pueden enviar los cambios a Githhub, pero con el pull se pueden traer los cambios de Github a la computadora. Cuando se hace un commit se recomienda escribir un mensaje descriptivo donde explique los cambios que se realizaron. Es como hacer un dibujo el cual aun no se termina y quieres que se quede guardado así para siempre entonces es donde se hace un commit, se pone un título al dibujo.

Repositorio remoto

Es como la copia de un proyecto que esta en otro lugar pero que es accesible y que las demás personas lo pueden ver para que también pueda formar parte del proyecto todos pueden ver los cambios que cada persona aporta y trabajar juntos en el mismo proyecto. Es algo parecido a una nube, se puede acceder desde cualquier dispositivo sin perder nada. Los comandos pull y push funcionarían como subir y bajar archivos.

¿Qué son las etiquetas en HTML?

Son las piezas fundamentales para construir páginas web. Son como instrucciones que le das al navegador para que sepa como mostrar el contenido. Hay muchas etiquetas diferentes para todo tipo de contenido: texto, imágenes, enlances, listas, tablas y muchos más, son esenciales para diseñar y organizar el contenido de una página web de manera efectiva y clara.

¿Cuáles hay y para qué sirven? Mínimo 10

1. <table>: Se utiliza para crear tablas en una página web. Envuelve todo el contenido de la tabla.
2. <tr>: Define una fila dentro de una tabla (<table>).
3. <th>: Define una celda de encabezado en una tabla. El texto dentro de esta etiqueta suele mostrar en negrita y centrado.
4. <td>: Define una celda de datos estándar en una tabla.
5. <ul>: Crea una lista desordenada (con números o letras).
6. <ol>: Crea una lista ordenada (con números o letras).
7. <li>: Define cada elemento dentro de una celda (<ul> o <ol>)
8. <form>: Se utiliza para crear formularios HTML, que se usan para recopilar información del usuario (como nombres, contraseñas, etc…)
9. <input>: Crea un control de entrada interactivo para formularios web. Puede ser un campo de texto, una casilla de verificación, un botón de radio, etc., dependiendo de su atributo type.
10. <button>: Crea un botón clicable en la página