readem.md 2024-05-05

Clase01: 30 Abril del 2024

Comandos visual code

- Windows: ==CTRL+SHIFT+P== : Abre el panel de control que tiene VSC.
- **Windows**: ==CTRL+P== : Abre panel para cambiar de diferente archivo que tengas en tu carpeta.
- **Windows**: ==CTRL+B== : La pestaña que te muestra tus archivos.
- **Windows**: ==CTRL+D== : Te permite cambiar el nombre de varias palabras/variables del mismo nombre.
- **Windows**: ==CTRL+F== : Te ayuda a buscar palbara/variable en el codigo.
- **windows**: ==CTRL+S== : Es la opcion para guardar.
- Windows: ==AIT + or -== : Si esta abierto pestañas te permite cambiar a otra.
- Windows: ==AIT + up or down== : Mueve toda la linea de acuerdo a la flecha.
- windows: ==SHIFT+ALT+ up or down== : Copia toda la linea a la siguiente.
- windows: ==SHIFT+ALT+A== : Pone para comentar una linea.
- windows: ==CTRL+K+C== : Pondra toda la linea como comentario.
- windows: ==CTRL+T== : Permite mostrar todos los simbolos.
- windows: ==CTRL+SPACE== : Te ayuda autocompletar una palabra.
- windows: ==CTRL+x== :Borra una linea.
- windows: ==CllCK en el archivo + F2== : Cambia el normbre del archivo.

Comandos de la terminal

- <u>pwd</u>: Me permite saber en que directorio estoy trabajando.
- touch: Para crear un archivo. Ej: touch readme.md
- <u>code</u> : Para crear archivos. *Ej: code readme.md*
- <u>touch "" >> texto2.txt</u>: Dentro de las comillas escribo lo que quiero que se escriba al crear el archivo -- > *EJ*: touch "hola mundo" >> texto2.txt

Git y Github

readem.md 2024-05-05



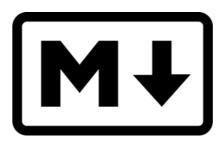
hecho en un conjunto de archivos es decir, un proyecto. Github es un servicio que almacena repositorios en sus servidores y control de versiones usando git.

Recomendaciones de otros comandos que tiene markdown

- Si queremos enumerar hacemos lo siguiente:
 - 1. Linea 1
 - 2. Linea 2
 - 1. Linea 1
 - 2. linea 2
 - 3. Linea 3
- Si queremos poner otro signo para ideas hacemos lo siguiente:
 - Linea 1
 - o Linea 2
 - Linea 1
 - linea 2
 - Linea 3

Clase03: 31 Abril del 2024

Markdown



Es un formato que nos permite generar archivos xtml y html utilizando una sintáxis bastante clara y sencilla y sobre todo fácil de escribir.

- markdown (para convertir a pdf mi markdown):
 - 1. Windows: CTRL+SHIFT+P
 - 2. Despues busco la opcion de "markdown export pdf"
 - 3. Se crea automaticamente un archivo pdf de tu markdown
- t * palabra o texto *
- ** palabra o texto **
- *** palabra o texto ***

readem.md 2024-05-05

- windows: ALT+9+6 (`) or windows: ALT+1+2+6 (~)
 - Nos permite poner una parte del codigo para analizar despues de (```) tenemos que poner el lenguaje de progrmacion
 - Ejemplos: 1. java public int sumar(){ int i=10; } 2. java public clas Hola{
 /* clase principal */ public class void main{ System.out.println(); } }
- Para hacer cuadros:

Columna 1	columna 2
А	В
С	D

- Comandos de git
 - 1. choco: Revisar en el internet.
 - 2. git init: Crea un carpeta oculta en tu carpeta de proyecto y va aparecer tus archivo con "U".
 - 3. <u>git status</u>: Te muestra que archivos has guardado --> (rojo: no esta guardado and verde: esta guardado).
 - 4. git commit: Para guradar el proyecto.
 - 5. <u>git checkout -b "Nombre de rama"</u>: Crea y cambia de rama.
 - 6. git branch: Crea una rama.
 - 7. git push: Subo todo al github.
 - 8. git pull: Nos ayuda a traer los cambios hechos en la nube.
 - 9. git clone url : Baja el archivo de algun proyecto.
 - 10. git ignore: Dentro del archivo puedes poner archivos q va ignorar o no va a guardar.
 - 11. git config --global user.name "nombre": Colocar nombre de ususario.
 - 12. <u>git config user.name</u>: Verificar si esta su nombre de ususario. (**Es lo mismo para el email solo cambia user.email**)
 - 13. <u>qit confiq --qlobal init.default branch "main"</u>: Cambiar de master a main.
 - 14. git commit -m Descripcion :Agregar cambios al repositorio local y una descripcion del cambio.
 - 15. git log: Muestra historial de commits. Si agregas "--oneline", aparece mas compacto.
 - 16. git add . : Agregar cambios al commit.
 - 17. git config --global core.editor : Asociar al vscode.
 - 18. <u>git branch -m nombre actual nombre nuevo</u> : Cmabiar nombre de rama sin estar en la rama a la que se va a cambiar.
 - 19. <u>git branch -d</u>: Eliminar rama (locales no publicadas).
 - 20. <u>git merge nombre-de-rama</u>: Fusionar rama con main.
 - 21. git remote: Ver nombre del repositorio remoto.
 - 22. <u>git fetch origin</u>: Crear una rama local con los cambios de la rama remota.
 - 23. git push origin -d: Eliminar rama remota.