

# Clase numero 1: Introduccion

---

En la primera clase de la semana comenzando visualizando el silabo de la materia, el cual se encuentra en el siguiente enlace: [Silabo 2024- Programacion II](#)

## Comandos Linux

- **pwd:** Sirve para observar el directorio en el que estamos.
  - **touch:** Sirve para crear un archivo vacio.
  - **code:** Sirve para ingresar al archivo.
- 

## Comandos Visual Studio Code

- **Paleta de comandos:** CTRL+shift+P
- **Apertura rapida:** CTRL+P
- **Alternar ventanas:** CTRL+B
- **Cursor de seleccion multiple:** CTRL+D
- **Copiar linea:** MAYUS+ALT+fecha arriba
- **Bloqueo de codigo de comentario:** MAYUS+ALT+A (COMENTARIO DE VARIAS LINEAS)
- **Bloqueo de codigo de comentario:** MAYUS+K+C (COMENTARIO DE una sola linea)
- **Retroceder o avanzar:** ALT+(+ o -)
- **Mostrar todos los simbolos:** CTRL+T
- **Sugerencias de activacion o de parametros:** CTRL\*ESPACIO o CTRL+MAYUS+ESPACIO

# Clase numero 2: Primer Taller

En este taller trabajamos con unos grupos formados previamente la clase anterior con los siguientes materiales:

- **Fideos**
  - **Malvaviscos**
  - **Cinta Masquin**
  - **Cinta metrica**
- 

En resumen el objetivo era construir una torre utilizando la menor cantidad de material posibles y con la mayor altura posible.



24mm f/1.75 1/60s ISO1545

2 / 4

## Markdown

Para crear archivos de este lenguaje utilizamos la extension **.md** en Markdown y se dio un repaso de los siguientes comandos basicos:

- #Titulo#
- \*Cursiva \*
- \*\* Negrita \*\*
- \*\*\* Negrita y Cursiva \*\*\*
- ==Texto resaltado==
- ~~Texto tachado~~

---

---

[!NOTE]

Este es un cuadro de texto, sirve para añadir notas. Utilizando ">" al inicio de una linea"

---

```
public class HolaMundo {  
    public static void main(String[] args) {  
        System.out.println("Hola mundo");  
    }  
}
```

---

```
int main{  
printf hola mundo  
return 0  
}
```

---

Link [buscador google](#)

![EPN] (img /uno2.png)

Hello	World
-------	-------

How	are you?
-----	----------

## GIT

- git --version (Comprueba si el git esta instalado)

## Tu Identidad

- git config global user.name "Example01"
- git config global user.email Example@hotmail.com

## Verificacion

- git config user.name
- git config user.email

## Inicio de control de versiones

- git init (Crear un directorio y abre el directorio en vscode)
- git brach -m (Renombrar la rama master a main)
- git status (da información sobre los archivos que han sido modificadoso agregados al área de preparación)
- git add (Este comando se utiliza para agregar cambios del directorio de trabajo al área, prepara los cambios para ser incluidos en el próximo commit.)
- git add NombreCarpeta NombreArchivo.ext
- git commit -m (Este comando registra los cambios que están en el área de preparación en el repositorio. La opción -m te permite agregar un mensaje que describe los cambios realizados en el commit.)

## Inicio de control de versiones - clonado

- git clone https://github.com/xyyyy/abc
- git clone https://github.com/xyyyy/abc miPropioNombre

## Visualizacion de archivos ignorados

- cat gitignore (Especifica patrones de archivos que deben ser ignorados)

## Otros comandos

- git push