CLASE 01

30-04-2024

Comandos básicos en Visual Estudio Code

Todos los comandos a continuacion son ejecutables en windows:

- **CTRL + SHIFT + P**: Abre la paleta de comandos.
- **CTRL + P**: Abre el buscador de archivos.
- **CTRL + B**: Abre y cierra la barra lateral.
- **CTRL + D**: Activa el multi-cursor.
- **SHIFT + ALT + UP / SHIFT + ALT + DOWN**: Seleccionar toda la linea de codigo.
- **SHIFT + ALT + A**: Comentar varias lineas de codigo.
- **CTRL + K + C**: Comentar una linea de codigo.
- **ALT + ---->**: Retroceder o avanzar.
- **CTRL + T**: Mostrar comandos.
- **CTRL + ESPACIO**: Sugerencias de activacion.

CLASE 02

01-05-2024

Markdown

Es un lenguaje de marcado, su extension es ".md", se usara para anotar los apuntes de la clases y se puede transformar a pdf. Tiene varias funcionalidades para resaltar de diversas formas las partes del texto.

```
Titulo #        ##          ###         ####        #####

**palabras en negrita**

*palabras en cursiva*

***palabras en negrita y cursiva***

==Texto resaltado==

~~Texto tachado~~
```

Colocar imagenes

```
![Nombre alternativo] (ruta de acceso de la imagen)
```

Colocar lineas de codigo

```
int main (){
    cout << "Esto es un ejemplo"<<endl;
}
```

GIT

Herramienta con la cual podremos subir nuestros codigos a repositorios en linea, que facilita el compartido de los mismos, a continuacion los comandos mas usados:

Identidad del usuario

```
git config --global user.name “pat_mic“
```

```
git config --global user.email pat_mic@hotmail.com
```

```
git config user.email
```

```
git config user.name
```

Control de versiones

```
git init
```

```
git status
```

```
git add .
```

```
git add NombreCarpeta/NombreArchivo.ext
```

```
git commit -m 'mensaje : initial project version'
```

Control de versiones - clonando

```
git clone https://github.com/xxyyy/abc
```

```
$ git clone https://github.com/xxyyy/abc miPropioNombre
Touch (crear archivos)
```

Quitar archivos del control

```
echo ”nombreArchivo.ext” >> .gitignore
```

```
echo ”*.txt” >> .gitignore
```

Ver archivos ignorados

```
cat .gitignore
```

Ver archivos ignorados

```
git add -f NombreArchivo.log
```

Crear y clonar

```
git init
```

```
git clone /ruta del repositorio
```

CLASE 03

06/05/2024

JAVA

- Origen

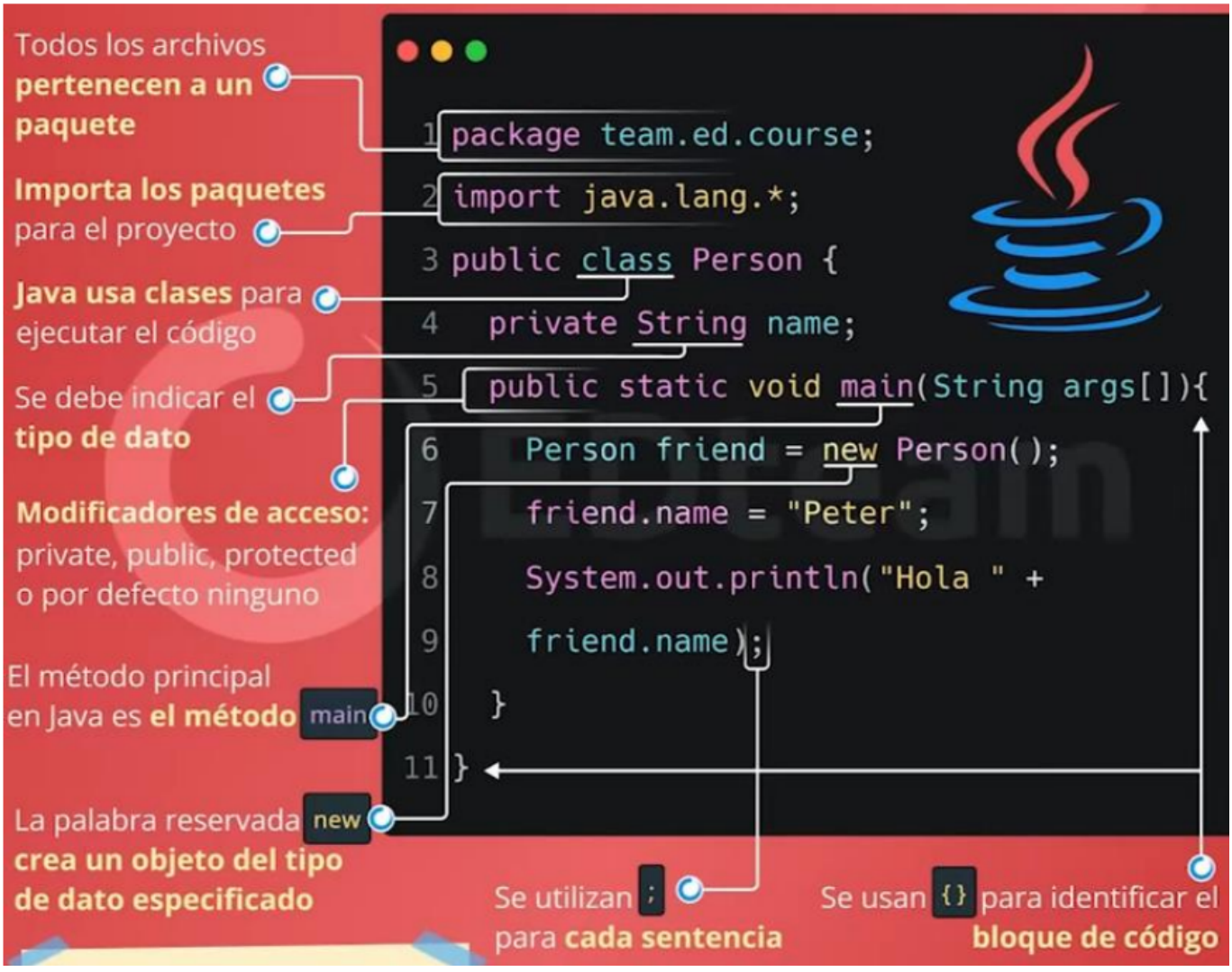
Como funciona:

1. Codigo
2. Compilacion
3. Bytecode
4. JVM
5. Multiplataforma.

Sabores de JAVA

- Java Micro Edition.
- Java Standar Edition.
- Java enterprise Edition.

Sintaxis de java



Tipos de lenguaje

- **Compilado:** Convierte el codigo a binarios que lee el S.O.
- **Interpretado:** Requiere de un programa que lea la intruccion del codigo en tiempo real.
- **Intermedio:** Se compila el codigo fuente a un lenguaje intermedio y este ultimo se ejecuta en una maquina virtual.

Estructurado vs O.O

Programacion estructurada

- La programacion estructurada tiene **funciones o procedimientos**.
- La programacion estructurada se maneja con **estructuras**.
- La programacion estructurada tiene **variables**.

Programacion O.O

- La programacion orientada a objetos tiene **metodos**.
- La programacion orientada a objetos se maneja con **clases**.
- La programacion orientada a objetos tiene **propiedades**.
- Las variables si existen, solo **dentro de los metodos**.

CLASE 07

13-05-2024

P.O.O

1. Conceptualizacion

Se refiere a la idea que se quiere realizar, se puede dibujar, prototipar, etc.

- Se necesita ponerle un nombre y concretar un significado.
- A los objetos se les puede atribuir dos cosas: caracteristicas o acciones:

Caracteristicas: propiedades, se debe poder almacenar, y debe almacenar informacion (descripcion precisa del objeto).

Acciones: metodos, son acciones, verbo, las acciones deben estar ligadas a las propiedades. Tienen parametros (Informacion necesaria para que se cumpla la accion de manera exitosal).

Ambitos:

- Public (+)
- Protect (-)

- Friendly (~)
- Protect

2. UML

(Lenguaje de modelado unificado) se usan las clases, todo lo conceptualizado se ubica dentro de una clase

Clase

- edad: float

+Tipocabello: string | ~ bailar (cancion:string, tiempomin:int, ritmo:string) : string | ~ tocar (Objeto:string, tiempoMin:int): boolean +Saltar (AlturaMts: int, cantidad:int):void

Se puede pedir que las acciones devuelvan un valor, que puede ser:

- void: No regresa nada
- String: que regrese una palabra o dato
- Boolean: regrese un valor verdadero o falso.

Eventos: Situaciones en la que interactua el objeto con otros objetos.

3. Codigo

```
public class mujer {
    private float edad;
    public boolean tieneojos;

    protected bailar (String cancion, TiempoMin int, String ritmo)
        return "sddsdsds";


}
```