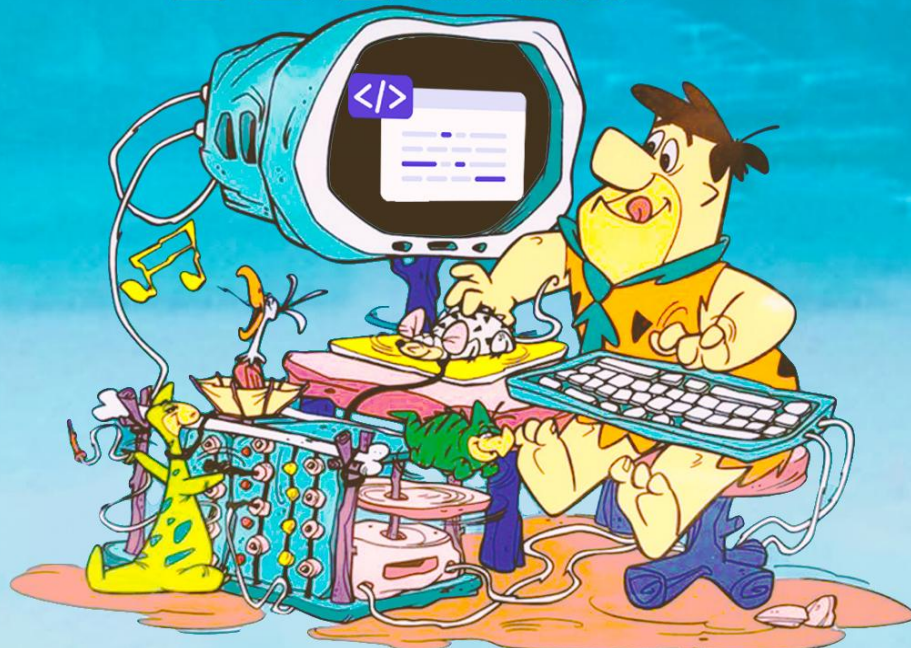


The JavaDabadev's

**UNIVERSIDAD
TECNOLÓGICA
NACIONAL**
Facultad Regional
San Rafael



Integrantes : Johana Martinez, Ciro Valentin Martinez, Araceli Pintos, Augusto Casado, Carolina Mamani Condori, Campos Margarita, Marcelo Boujon, David Esteche.

2) Página web que contenga los archivos columnas.html y columnas.css, añadir la etiqueta viewport y el enlace a la hoja de style de nombre columnas.css

```
EXPLORER
PÁGINA WEB
  css
    columnas.css
    columnas.html
columnas.html
columnas.css

columnas.html > html > body > main.main
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <meta name="viewport" content="width=device-width, initial-scale=1.0">
6    <link rel="stylesheet" href="css/columnas.css">
7    <title>Document</title>
8  </head>
9  <body>
10   <menu class="menu">
11     25%
12   </menu>
13
14   <main class="main">
15     75%
16   </main>
17
18 </body>
19
20 </html>
```

3) Realizar 1 ejemplo de Encapsulación, Abstracción, Herencia y Polimorfismo.

Encapsulamiento :

```
class Coche {  
  - marca: String  
  - modelo: String  
  - año: int  
  + obtenerMarca(): String  
  + obtenerModelo(): String  
  + obtenerAño(): int  
  + establecerMarca(m: String): void  
  + establecerModelo(mod: String): void  
  + establecerAño(a: int): void  
}
```

Abstracción :

```
abstract class ProcesoCompra {  
  + agregarProducto(producto: Producto): void  
  + eliminarProducto(producto: Producto): void  
  + calcularTotal(): double  
  + realizarPago(): void  
  + confirmarPedido(): void  
}
```

```
class CarritoCompra {  
  + agregarProducto(producto: Producto): void  
  + eliminarProducto(producto: Producto): void  
  + calcularTotal(): double  
}
```

```
class Pedido {  
  + confirmarPedido(): void  
  + realizarPago(): void  
}
```

```
ProcesoCompra <|-- CarritoCompra  
ProcesoCompra <|-- Pedido
```

Herencia :

```
class FiguraGeometrica {  
  - base: double  
  - altura: double  
  + calcularArea(): double  
}
```

```
class Triangulo {  
  + calcularArea(): double  
}
```

```
class Rectangulo {  
  + calcularArea(): double  
}
```

```
FiguraGeometrica <|-- Triángulo  
FiguraGeometrica <|-- Rectángulo
```

Polimorfismo :

```
interface Reproducible {  
  + reproducir(): void  
}
```

```
class Video implements Reproducible {  
  + reproducir(): void  
}
```

```
class Audio implements Reproducible {  
  + reproducir(): void  
}
```

```
Reproducible <|-- Video  
Reproducible <|-- Audio
```