Aplicaciones Móviles Multiplataforma

Ing. Walter Mauro Moncada Rubio

SEMANA 02

Manejo de Clases en DART

Tecsup

Objetivos

- Comprender la pregunta: ¿Cómo se manejan las clases en DART?
- Analizar a LOS CONSTRUCTORES.
- Aprender a como usar GETTER y SETTERS.
- Comprender como funciona una CLASE ABSTRACTA

Tecsup

¿Cómo funcionan las clases?

```dart
void main(){


   final wolverine = new Heroe();

```

Esta es la manera de como crear una instancia, considerando que en Flutter se evita utilizar new con la finalidad de manejar un código más limpio

```dart
   wolverine.nombre = 'Logan';
   wolverine.poder = 'Regeneración';

   print(wolverine);

}

class Heroe{

   String? nombre;
   String? poder;

}
```

Se puede otorgar un valor a la variable, todo partiendo de la creación de la clase pero…

Observamos un "?" en la declaración de la variable

¿A qué se debe?

Procedamos a imprimir datos

```
@override
String toString(){
  return 'nombre: $nombre, poder: $poder';
}
```

Procedemos primero a agregar @override para sobrescribir la salida de la cadena (buena práctica)

Y podemos dar salida de los valores asignado en las capturas anteriores. Sin embargo lo correcto es lo siguiente

```
@override
String toString(){
  return 'nombre: $this.nombre, poder: $this.poder';
}
```

- Right?
- Nope?
- Any idea?

Tecsup

¿Y el constructor?

Sabemos que su finalidad es inicializar la clase. Pero hay algunos detalles que debemos considerar en DART

```
Heroe(String vnombre, String vpoder){
  this.nombre = vnombre;
  this.poder = vpoder;
}
```

Podríamos hacer esto
¿?

En DART suele ser mejor opción de esta manera

```
final wolverine = new Heroe('Logan','Regeneracion');
```

```
class Heroe{

  String nombre;
  String poder;

  Heroe(this.nombre, this.poder);
```

Tecsup

Ahora como sería

Mandar los argumentos por nombre…

```
final wolverine = new Heroe(nombre: 'Logan',poder: 'Regeneracion');
```

```
class Heroe{

  String nombre;
  String poder;

  Heroe({required this.nombre, required this.poder});
```

Tecsup

¿Constructores con NOMBRE?

Si necesitamos crear un constructor en base a un mapa

```
final dataJson = {

     'nombre': 'Charles X Javier',
     'poder': 'Telequinesis'
};
```

```
String? nombre;
String? poder;
```

Esta sería la solución en la clase

```
class Heroe{

  String? nombre;
  String? poder;

  Heroe({required this.nombre, required this.poder});

  Heroe.fromdataJson(Map <String,String> json){
    this.nombre = json['nombre']!;
    this.poder = json['poder']!;
  }
```

Tecsup

¿Constructores con NOMBRE?

Resultado del proceso anterior…

```
final dataJson = {

      'nombre': 'Charles X Javier',
      'poder': 'Telequinesis'
};

final xmen = new Heroe.fromdataJson( dataJson);

print(xmen);

}
```

Console

```
nombre: Charles X Javier, poder: Telequinesis
```

Tecsup

¿Podemos realizar algo mejor?

```dart
class Heroe{

  String nombre;
  String poder;

  Heroe({required this.nombre, required this.poder});

  Heroe.fromdataJson(Map <String,String> json):
    this.nombre = json['nombre'] ?? 'No tiene nombre',
    this.poder = json['poder'] ?? 'No tiene poder';

  @override
  String toString(){
    return 'nombre: ${this.nombre}, poder: ${this.poder}';
  }

}
```

Tecsup