Sección 2: Introducción a Dart

 5. Introducción a la sección

 6. Temas puntuales de la sección

 7. ¿Por qué Dart?

VIDEO 8. Hola Mundo

VIDEO 9. Tipos de datos - Números y Strings

VIDEO 10. Tipo de dato - Booleanos y condiciones

VIDEO 11. Tipo de dato - Lista

VIDEO 12. Tipo de dato - Map

VIDEO 13. Funciones en Dart

VIDEO 14. Clases en Dart

VIDEO 15. Constructores con nombre

VIDEO 16. Getters y Setters

VIDEO 17. Clases abstractas

VIDEO 18. Extends

VIDEO 19. Mixins

VIDEO 20. Futures

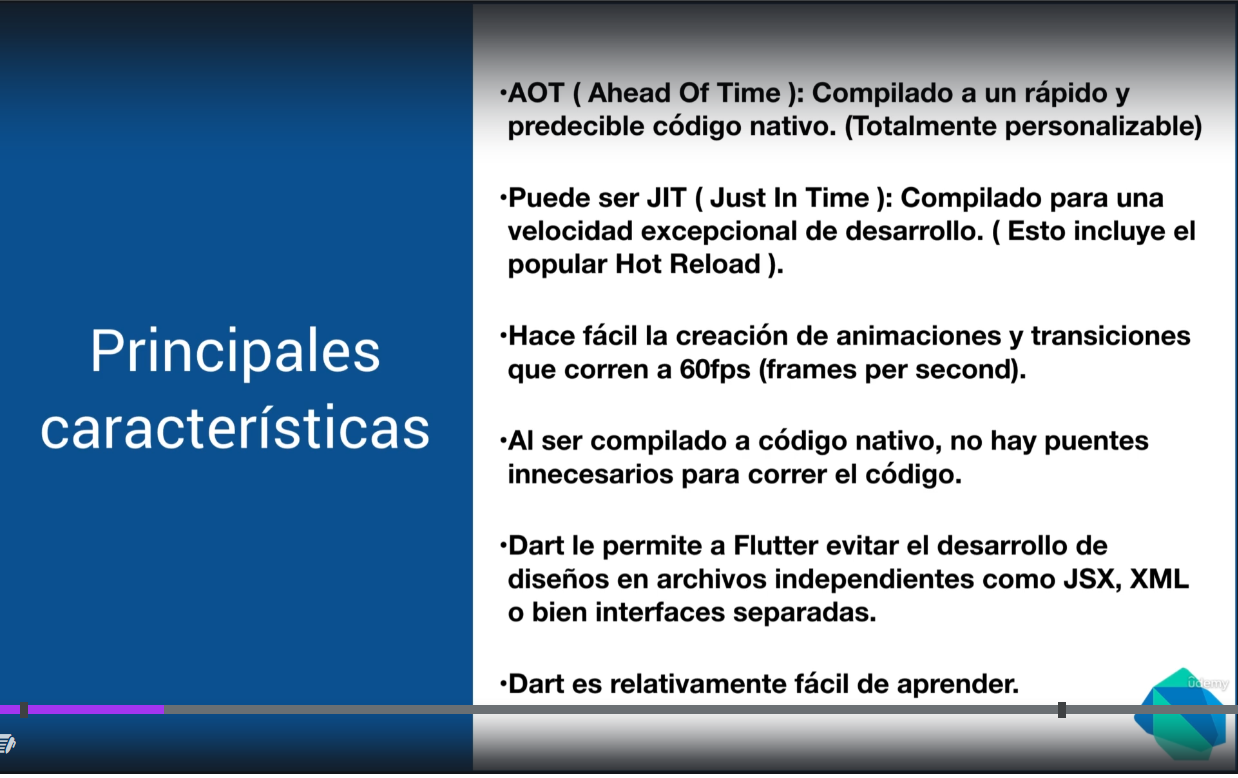
VIDEO 21. Async - Await

Temas puntuales de la sección

En esta sección tocaremos temas sobre:

* ¿Por qué Dart?
* Nuestro "Hola Mundo" en Dart
* Tipado de datos en Dart
* Listas
* Mapas
* Funciones
* Clases
* Mixins
* Constructores
* Getters
* Setters
* Extender una clase
* Futures
* Async
* Await

VIDEO 7. ¿Por qué DART?



Video8. HOLA MUNDO

<https://dartpad.dartlang.org/?null_safety=true>

VIDEO 9. Tipos de datos - Números y Strings

void main*() {* // Strings  
 final String nombre = 'Tony';  
 final apellido = 'Stark';  
  
// nombre = 'Peter';  
  
 print*(*'$nombre $apellido'*)*;  
  
  
// Números  
 int empleados = 10;  
 double salario = 1856.25;  
  
 print*(* empleados *)*;  
 print*(* salario *)*;  
*}*

VIDEO 10. Tipo de dato - Booleanos y condiciones

void main*() {* bool activado = true;  
 print*(*activado*)*;  
   
   
 if *(* !activado *) {* print*(*'El motor esta funcionando'*)*;  
 *}* else *{* print*(*'Está apagado'*)*;  
 *}  
   
   
   
}*void main*() {* bool? isActive = null;  
  
 if *(* isActive == null *) {* print*(* 'isActive es null' *)*;  
 *}* else *{* print*(* 'No es null' *)*;  
 *}  
  
  
}*

VIDEO 11. Tipo de dato – Lista

void main*() {* List*<*int*>* numeros = *[*1,2,3,4,5*]*;  
 print*(* numeros *)*;  
   
 numeros.add*(*6*)*;  
 print*(* numeros *)*;  
   
   
 // Tamaño fijo  
 List masNumeros = List*(*10*)*;  
 print*(* masNumeros *)*;  
// masNumeros.add(1); Esto no es correcto  
 masNumeros*[*0*]* = 1;  
 print*(* masNumeros *)*;  
   
*}*void main*() {* List*<*int*>* numeros = *[*1,2,3,4,5,6,7,8,9,10*]*;  
 numeros.add*(*11*)*;  
 print*(* numeros *)*;  
  
 final masNumeros = List.generate*(*100, *(*int index*)* => index *)*;  
  
 print*(*masNumeros*)*;  
*}*

VIDEO 12. Tipo de dato – Map

void main*() {* String propiedad = 'soltero';  
   
   
 Map*<*String, dynamic*>* persona = *{* 'nombre' : 'Carlos',  
 'edad' : 32,  
 'soltero': true  
 *}*;  
   
 print*(* persona*[*'nombre'*] )*;  
 print*(* persona*[*'edad'*] )*;  
 print*(* persona*[* propiedad *] )*;  
   
   
 Map*<*int, String*>* personas = *{* 1: 'Tony',  
 2: 'Peter',  
 9: 'Strange'  
 *}*;  
   
 personas.addAll*( {* 4: 'Banner' *} )*;  
   
 print*(* personas *)*;  
 print*(* personas*[*2*] )*;  
   
   
*}*void main*() {*// Map persona = {  
// 'nombre': 'Fernando',  
// 'edad': 35,  
// 'soltero': false,  
// true: false,  
// 1: 100,  
// 2: 500  
// };  
  
 Map*<*String, dynamic*>* persona = *{* 'nombre': 'Fernando',  
 'edad': 35,  
 'soltero': false,  
 *}*;  
  
 persona.addAll*({* 'segundoNombre': 'Juan' *})*;  
  
 print*(* persona *)*;  
*}*

VIDEO 13. Funciones en Dart

void main*() {* String mensaje = saludar2*(* nombre: 'Fernando', texto: 'Hola,' *)*;  
   
 print*(*mensaje*)*;  
   
*}*String saludar*({* String texto, String nombre *}) {*// print('Hola');  
 return '$texto $nombre';  
*}*String saludar2*({* String texto, String nombre *})* => '$texto $nombre';  
  
  
void main*() {* final nombre = 'Fernando';  
  
// saludar( nombre, 'Greetings' );  
 saludar2*(* nombre: nombre, mensaje: 'Greetings' *)*;  
  
*}*void saludar*(* String nombre, *[* String mensaje = 'Hi' *]) {* print*(*'$mensaje $nombre'*)*;  
*}*void saludar2*({* required String nombre,  
 required String mensaje,  
*}) {* print*(*'$mensaje $nombre'*)*;  
*}*

VIDEO 14. Clases en Dart

void main*() {* final wolverine = new Heroe*(* poder: 'Regeneración',  
 nombre: 'Logan'  
 *)*;  
   
 print*(* wolverine *)*;  
   
*}*class Heroe *{* String nombre;  
 String poder;  
   
// Heroe({ String nombre = 'Sin Nombre', String poder }) {  
// this.nombre = nombre;  
// this.poder = poder;   
// }  
   
 Heroe*({* this.nombre, this.poder *})*;  
   
 String toString*()* => 'nombre: $nombre - poder: $poder';  
   
*}*void main*(){* final wolverine = new Heroe*(*nombre:'Logan', poder: 'Regeneración'*)*;  
  
// wolverine.nombre = 'Logan';  
// wolverine.poder = 'Regeneración';  
  
 print*(* wolverine *)*;  
  
*}*class Heroe *{* String nombre;  
 String poder;  
  
 Heroe*({* required this.nombre,  
 required this.poder  
 *})*;  
  
// Heroe( String pNombre ) {  
// this.nombre = pNombre;  
// }  
  
  
 String toString*() {* return 'Heroe: nombre: *${*this.nombre*}*, poder: *${* this.poder *}*';  
 *}  
}*

VIDEO 15. Constructores con nombre

import 'dart:convert';  
  
void main*() {*// final wolverine = new Heroe('Logan', 'Regeneración');  
 final rawJson = '{ "nombre": "Logan", "poder":"Regeneración" }';  
 Map parsedJson = json.decode*(* rawJson *)*;  
   
// print( parsedJson );  
   
 final wolverine = new Heroe.fromJson*(* parsedJson *)*;  
   
 print*(*wolverine.nombre*)*;  
 print*(*wolverine.poder*)*;  
   
*}*class Heroe *{* String nombre;  
 String poder;  
   
 Heroe*(* this.nombre, this.poder *)*;  
   
 Heroe.fromJson*(* Map parsedJson *) {* nombre = parsedJson*[*'nombre'*]*;  
 poder = parsedJson*[*'poder'*]*;  
   
 *}  
   
   
}*void main*(){* final rawJson = *{* 'nombre': 'Tony Stark',  
 'poder': 'Dinero'  
 *}*;  
  
// final ironman = new Heroe( nombre: rawJson['nombre']!, poder: rawJson['poder']! );  
  
 final ironman = Heroe.fromJson*(* rawJson *)*;  
  
 print*(*ironman*)*;  
  
// final wolverine = new Heroe(nombre:'Logan', poder: 'Regeneración');  
// print( wolverine );  
  
*}*class Heroe *{* String nombre;  
 String poder;  
  
 Heroe*({* required this.nombre,  
 required this.poder  
 *})*;  
  
 Heroe.fromJson*(* Map*<*String, String*>* json *)*:  
 this.nombre = json*[*'nombre'*]*!,  
 this.poder = json*[*'poder'*]* ?? 'No tiene poder';  
  
  
  
 String toString*() {* return 'Heroe: nombre: *${*this.nombre*}*, poder: *${* this.poder *}*';  
 *}  
}*

VIDEO 16. Getters y Setters

void main*() {* final cuadrado = new Cuadrado*()*;  
   
 cuadrado.lado = 10;  
   
   
 print*(* cuadrado *)*;  
 print*(* 'área: *${* cuadrado.area *}*' *)*;  
   
*}*class Cuadrado *{* double \_lado;  
// double \_area;  
   
   
 set lado*(* double valor *) {* if *(* valor <= 0 *) {* throw*(*'El lado no puede ser menor o igual a 0'*)*;  
 *}* \_lado = valor;  
   
 *}* double get area => \_lado \* \_lado;  
   
   
 toString*()* => 'Lado: $\_lado';  
   
*}*import 'dart:math' as math;  
  
void main*() {* final cuadrado = new Cuadrado*(* 2 *)*;  
  
 cuadrado.area = 100;  
  
 print*(* 'area: *${* cuadrado.calculaArea*() }*' *)*;  
  
 print*(* 'lado: *${* cuadrado.lado *}*' *)*;  
 print*(* 'area get: *${* cuadrado.area *}*' *)*;  
  
  
  
*}*class Cuadrado *{* double lado; // lado \* lado  
  
  
 double get area *{* return this.lado \* this.lado;  
 *}* set area*(* double valor *) {* this.lado = math.sqrt*(*valor*)*;  
 *}* Cuadrado*(* double lado *)*:  
 this.lado = lado;  
  
 double calculaArea*() {* return this.lado \* this.lado;  
 *}  
}*

VIDEO 17. Clases abstractas

VIDEO 18. Extends

VIDEO 19. Mixins

VIDEO 20. Futures

VIDEO 21. Async - Await