Temas puntuales de la sección

En el primer elemento tocaremos temas sobre:

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



**¿Qué es Dart?**

Dart es un lenguaje open source desarrollado en Google con el objetivo de permitir a los desarrolladores utilizar un lenguaje orientado a objetos y con análisis estático de tipo. Desde la primera versión estable en 2011, Dart ha cambiado bastante, tanto en el lenguaje en sí como en sus objetivos principales. Con la versión 2.0, el sistema de tipo de Dart pasó de opcional a estático, y desde su llegada, Flutter (explicamos qué es más adelante) se ha convertido en el principal objetivo del lenguaje.

**¿Qué hace a Dart especial?**

A diferencia de muchos lenguajes, Dart se diseñó con el objetivo de hacer el proceso de desarrollo lo más cómodo y rápido posible para los desarrolladores. Por eso, viene con un conjunto bastante extenso de herramientas integrado, como su propio gestor de paquetes, varios [compiladores/transpiladores](https://dartpad.dartlang.org/), un analizador y formateador. Además, la máquina virtual de Dart y la compilación Just-in-Time hacen que los cambios realizados en el código se puedan ejecutar inmediatamente.

Una vez en producción, el código se puede compilar en lenguaje nativo, por lo que no es necesario un entorno especial para ejecutar. En caso de que se haga desarrollo web, Dart se transpila a JavaScript.

En cuanto a la sintaxis, la de Dart es muy similar a lenguajes como JavaScript, Java y C ++, por lo que aprender Dart sabiendo uno de estos lenguajes es cuestión de horas.

Además, Dart consta de un gran apoyo para la asincronía, y trabajar con generadores e iterables es extremadamente sencillo.

**¿Dónde puedo utilizar Dart?**

Dart es un lenguaje de propósito general, y lo puedes utilizar casi para cualquier cosa:

* En aplicaciones web, utilizando la librería de arte: HTML y el transpilador para transformar el código en Dart en JavaScript, o utilizando frameworks como AngularDart.
* En servidores, utilizando las librerías de arte: http y arte: io. También hay varios frameworks que se pueden utilizar, como por ejemplo Aqueduct.
* En aplicaciones de consola.
* En aplicaciones móviles gracias a Flutter.