Dart Programming Language



What is Dart?

Dart is a general-purpose, high-level modern programming language which is originally developed by Google. It is the new programming language which is emerged in 2011, but its stable version was released in June 2017. Dart is not so popular at that time, but It gains popularity when it is used by the Flutter.

Dart is a dynamic, class-based, object-oriented programming language with closure and lexical scope. Syntactically, it is quite similar to [Java](https://www.javatpoint.com/java-tutorial), [C](https://www.javatpoint.com/c-programming-language-tutorial), and JavaScript. If you know any of these programming languages, you can easily learn the Dart programming language.

Dart is an open-source programming language which is widely used to develop the mobile application, modern web-applications, desktop application, and the [Internet of Things](https://www.javatpoint.com/iot-tutorial) (IoT) using by Flutter framework. It also supports a few advance concepts such as interfaces, mixins, abstract classes, refield generics, and type interface. It is a compiled language and supports two types of compilation techniques.

* **AOT (Ahead of Time) -** It converts the Dart code in the optimized JavaScript code with the help of the dar2js compiler and runs on all modern web-browser. It compiles the code at build time.
* **JOT (Just-In-Time) -** It converts the byte code in the machine code (native code), but only code that is necessary.

History

Dart was revealed for the first time in the GOTO conference in the month of 10th - 12th October 2011 at Aarhus, Denmark. It is initially designed by the **Lars bark and Kespar** and developed by Google.

The first version 1.0 of Dart was released on November 14th, 2013, intended as a replacement of [JavaScript](https://www.javatpoint.com/javascript-tutorial).

In July 2014, the first edition of Dart language was approved by Ecma International approved at its 107th General Assembly.

The recent version Dart 3.6 is supplemented with the extension method, which enables us to add any type of functionality.

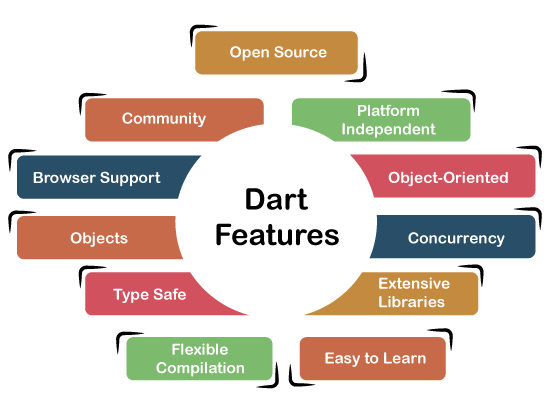
Why Dart?

We define the characteristics of Dart in the following point.

* Dart is a platform-independent language and supports all operating systems such as Windows, Mac, Linux, etc.
* It is an open-source language, which means it available free for everyone. It comes with a BSD license and recognized by the ECMA standard.
* It is an object-oriented programming language and supports all features of oops such as inheritance, interfaces, and optional type features.
* Dart is very useful in building real-time applications because of its stability.
* Dart comes with the dar2js compiler which transmits the Dart code into JavaScript code that runs on all modern web browser.
* The stand-alone Dart VM permits Dart code to run in a command-line interface environment.

# Dart Features

The Dart is an object-oriented, open-source programming language which contains many useful features. It is the new programming language and supports an extensive range of programming utilities such as interface, collections, classes, dynamic and optional typing. It is developed for the server as well as the browser. Below is the list of the important Dart features.



### Open Source

Dart is an open-source programming language, which means it is freely available. It is developed by Google, approved by the ECMA standard, and comes with a BSD license.

### Platform Independent

Dart supports all primary operating systems such as [Windows](https://www.javatpoint.com/windows), [Linux](https://www.javatpoint.com/linux-tutorial), Macintosh, etc. The Dart has its own Virtual Machine which known as Dart VM, that allows us to run the Dart code in every operating system.

### Object-Oriented

Dart is an object-oriented programming language and supports all oops concepts such as classes, inheritance, interfaces and optional typing features. It also supports advance concepts like mixin, abstract, classes, reified generic, and robust type system.

### Concurrency

Dart is an asynchronous programming language, which means it supports multithreading using Isolates. The isolates are the independent entities that are related to threads but don't share memory and establish the communication between the processes by the message passing. The message should be serialized to make effective communication. The serialization of the message is done by using a snapshot that is generated by the given object and then transmits to another isolate for desterilizing.

### Extensive Libraries

Dart consists of many useful inbuilt libraries including SDK (Software Development Kit), core, [math](https://www.javatpoint.com/math), async, math, convert, [html](https://www.javatpoint.com/html-tutorial), IO, etc. It also provides the facility to organize the Dart code into libraries with proper namespacing. It can reuse by the import statement.

### Easy to learn

As we discussed in the previous section, learning the Dart is not the Hercules task as we know that Dart's syntax is similar to [Java](https://www.javatpoint.com/java-tutorial), [C#](https://www.javatpoint.com/c-sharp-tutorial), [JavaScript](https://www.javatpoint.com/javascript-tutorial), [kotlin](https://www.javatpoint.com/kotlin-tutorial), etc. if you know any of these languages then you can learn easily the Dart.

### Flexible Compilation

Dart provides the flexibility to compile the code and fast as well. It supports two types of compilation processes, AOT (Ahead of Time) and JIT (Just-in-Time). The Dart code is transmitted in the other language that can run in the modern web-browsers.

### Type Safe

The Dart is the type safe language, which means it uses both static type checking and runtime checks to confirm that a variable's value always matches the variable's static type, sometimes it known as the sound typing.

Although types are required, type annotations are optional because of type interference. This makes it code more readable. The other advantage to being type-safe language is, when we change the part of code, the system warns us about that modification that we have modified earlier.

### Objects

The Dart treats everything as an object. The value which assigns to the variable is an object. The functions, numbers, and strings are also an object in Dart. All objects inherit from **Object** class.

### Browser Support

The Dart supports all modern web-browser. It comes with the **dart2js** compiler that converts the Dart code into optimized JavaScript code that is suitable for all type of web-browsers.

### Community

Dart has a large community across the world. So if you face problem while coding then it is easy to find help. The dedicated developers' team is working towards enhancing its functionality.

Here we have discussed essential features of the Dart language. We will more concepts of Dart language in upcoming tutorials.

# Dart Control Flow Statement

The **control statements** or **flow of control statements** are used to control the flow of Dart program. These statements are very important in any programming languages to decide whether other statement will be executed or not. The code statement generally runs in the sequential manner. We may require executing or skipping some group of statements based on the given condition, jumps to another statement, or repeat the execution of the statements.

Categories of Flow Statement

In Dart, Control flow statement can be categorized mainly in three following ways.

* Decision-making statements
* Looping statements
* Jump statements

