Frameworks, librerías, IDEs y editores de texto

DigitalHouse >
Coding School

Certified Tech Developer
The Ultimate Degree

Índice

1 | Frameworks

Un framework **es un patrón o esquema que ayuda a la programación** a estructurar el código, ahorrando tiempo y esfuerzos a los programadores.

Frameworks

Un framework es un conjunto de herramientas que nos facilitan el desarrollo de software. Para poder lograrlo incluyen implementos como los siguientes:

- APIs
- Librerías
- Herramientas de depuración
- Edición
- Prototipado
- Programas de soporte

Flask

Express JS

spring MVC

Vue.js

laravel

Frameworks

Los frameworks no están ligados necesariamente a un lenguaje concreto, aunque así sea en muchas ocasiones. Por ejemplo, en Ruby on Rails, “Ruby” es el lenguaje de programación y “Rails”, el framework.

Sin embargo, no existen impedimentos para definir el mismo framework para lenguajes diferentes.
También es posible que el framework defina una estructura para una aplicación completa, o bien solo se centre en un aspecto de ella.

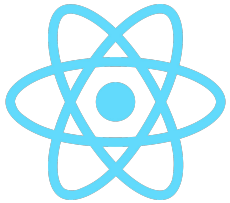
2 | Librerías

Una librería no es más que **un conjunto de código que alguien ha realizado** para que podamos reutilizar dentro de nuestros proyectos.

Librerías

El objetivo de una librería no es otro que hacer más fácil y rápido el desarrollo de ciertas funciones dentro de nuestro programa o aplicación.

Normalmente las librerías están enfocadas a solucionar problemas concretos. Es decir, no nos brindan una estructura para nuestro proyecto, pero sí van a ayudar a resolver funcionalidades específicas.



3 Diferencias entre frameworks y librerías

Librerías vs. Frameworks

Una **librería** es un código escrito previamente, ya utilizado por otros desarrolladores, listo para que lo utilicemos y pretende hacernos la vida más fácil y su trabajo más rápido.

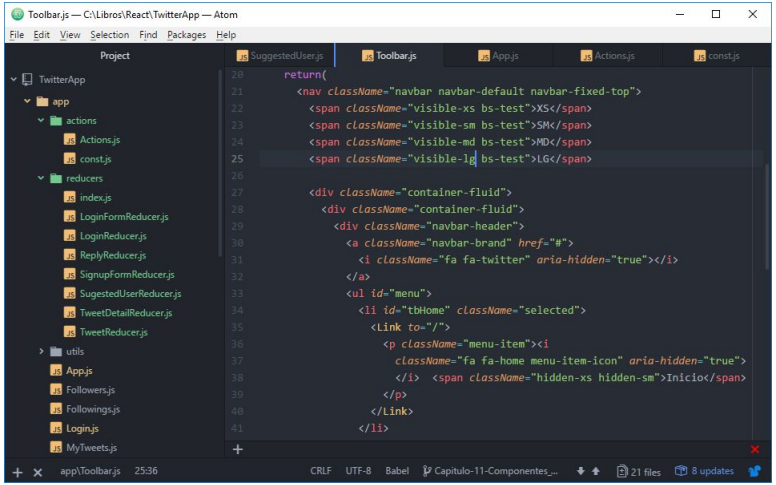
Un **framework** es un esqueleto. Podría decirse que es como el marco de una casa. La estructura está predeterminada, y el trabajo del desarrollador es llenar los vacíos con su código.

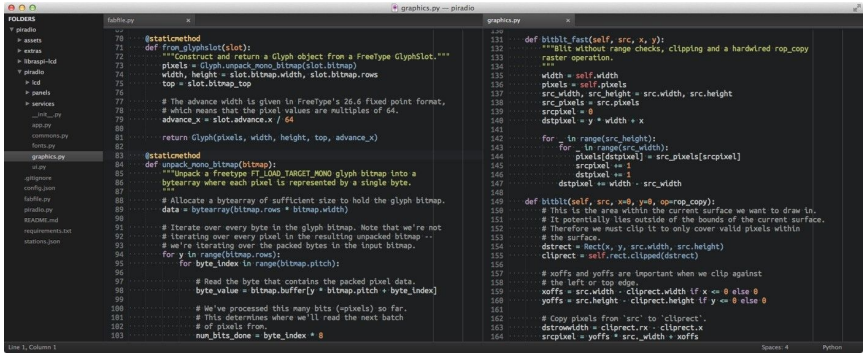
4 | Editores de texto

“Los editores de texto **se crearon para mostrar el código de una forma agradable y realizaban algunas acciones muy simples**. Sin embargo, los editores de texto modernos, siguen agregando capacidades que solo los IDEs tenían.”

Editores de texto

Los editores de texto son herramientas mucho más simples y compactas. Proporcionan un entorno de desarrollo simple. Los editores de texto tienen la peculiaridad de trabajar con archivos de texto y carpetas, es decir, al abrir una carpeta podemos trabajar con todo lo que hay dentro.





Existen editores sofisticados como Atom, Sublime Text y Brackets, los cuales muestran el código de una forma pintoresca y atractiva.

5 | IDEs

Los IDEs, a diferencia de los editores de texto, no trabajan con archivos y carpetas. En su lugar, **emplean el concepto de proyectos**.

IDEs

Un **IDE o *Integrated Development Environment*** (entornos de desarrollo integrado) posee muchas más herramientas que se integran en un mismo programa.

Los IDEs tienen como principal característica que **no trabajan con archivos y carpetas.** En su lugar, **emplean el concepto de proyectos**.

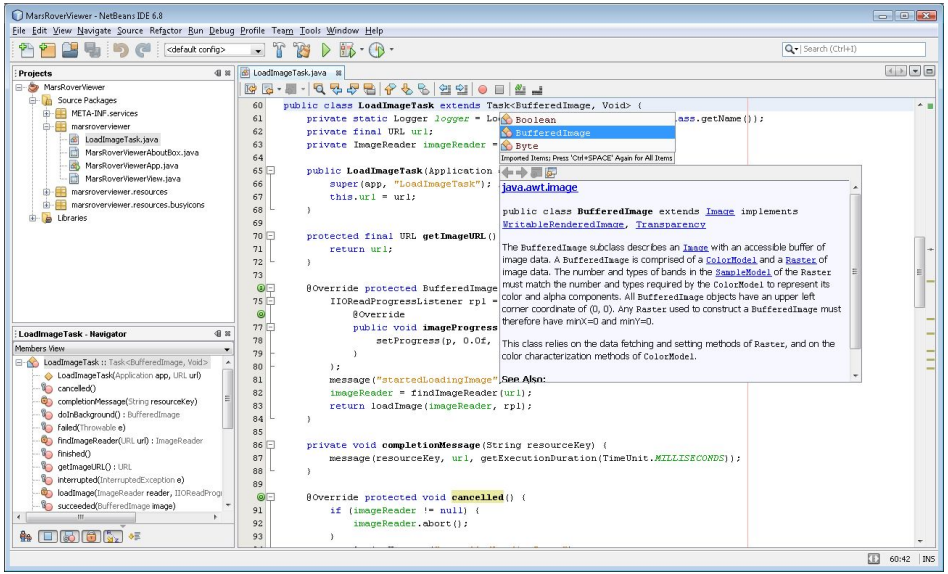
Un proyecto es una carpeta en el disco duro, pero tiene la diferencia que el IDE **crea archivos adicionales al código para optimizar la experiencia del usuario**. En estos archivos puede tener configuraciones de ejecución, deploy, tipo de proyecto, etc.

Debido a que los IDEs son plataformas muy complejas, es posible hacer un sinfín de cosas, y los plugins que ofrecen son prácticamente ilimitados.

IDEs

Las características que más resaltan de un IDE son:

- Debugger en tiempo real.
- Visualiza gráficamente casi cualquier cosa, desde XML, JSON, UML, bases de datos, interfaces gráficas, etc.
- Ayuda en tiempo real.




DigitalHouse>
Coding School