

Tecnologías de interés (2022)

Git

Para entender lo mejor posible el funcionamiento, te recomiendo el libro [Pro Git](#). También está en [español](#).

Enfocate en los primeros capítulos: Basics, Branching, y después saltá a Tools y Customizing, con el tiempo.

Si querés revisar los comandos por arriba, lo mejor es [A Visual Git Reference](#). También está en [español](#)

Linux

Es importante poder realizar acciones básicas en la línea de comandos: Filesystem (pwd, ls, cd, mkdir, cp, mv, rm), File Operations (cat, head, tail, grep), Processes (top, ps), Search (find, grep), Network Operations (ping, ssh).

Podés encontrar lo que necesites en un curso como [Introduction to Linux](#)

Sería bueno que conozcas algo de [Bash scripting](#), sed, awk.

Backend

Perl

Para tener un panorama inicial podés comenzar con [Learn Perl in about 2 hours 30 minutes - Sam Hughes](#)

O si lo preferís, descargar un PDF con contenidos similares: [Perl - Notes for Professionals](#)

Para probar código en la web podés usar [Execute Perl Online](#)

Si estás buscando un libro (algo ya más robusto y extenso) revisá [Modern Perl](#)

Python 3

Podés arrancar con [Python 3 básico](#), haciendo foco sobre funciones, estructuras de datos, [Modules](#) y [Packages](#).

Si preferís videos, repasá esta playlist de [Python for Beginners](#)

Si preferís algo más interactivo probá con [LearnPython](#)

Django

Una vez que tengas los conceptos previos, sería bueno ver algo de [Django](#).

Podés probar con un curso online basado en una versión previa: [Try Django 1.11 // Python Web Development](#)

MySQL

Repasá los contenidos de *beginner* en [SQL Tutorial](#) y si los tenés claros, pasá a los de *intermediate*.

Si preferís algo más práctico, pero básico, probá con [Select Star SQL](#)

Web

HTTP

Es importante que conozcas el protocolo [HTTP](#). También en [español](#)

Tené una idea de los distintos [Methods](#) (GET, POST, entre los más usados) y del funcionamiento de las [Cookies](#).

Sobre el ciclo de petición-respuesta, tal vez uno de estos videos te brinde un mejor panorama:

- [The Journey of an HTTP request to the Backend](#)
- [Request-Response Cycle](#)

HTML

Revisá los contenidos en orden y profundizá en aquellos donde tengas dudas. Más allá de tener una idea de los elementos disponibles, lo importante son los **formularios** y **controles**.

- [HTML Beginner Tutorial](#)
- [HTML Intermediate Tutorial](#)
- [HTML Advanced Tutorial](#)

CSS

Para arrancar con CSS, su estructura y cómo trabaja, probá con alguna de estas alternativas:

- [CSS first steps](#)
- [The Ultimate Guide to Learning CSS](#)
- [Learn CSS](#)

Algunos conceptos en los que enfocarte: Selectors, Box Model, Layout, Styling Text, Styling Boxes, Units, Colors and Gradients, Pseudo-classes and Pseudo-elements, Flexbox, Grid, Spacing, Media Queries, Specificity.

Para profundizarlos revisá [Learn to Code Advanced HTML & CSS](#)

Javascript

Para un repaso de lo básico hasta tópicos avanzados con explicaciones simples pero detalladas seguí [The Modern JavaScript Tutorial](#). También está en [español](#)

Para aclarar ciertos conceptos críticos, te recomiendo [JavaScript: Understanding the Weird Parts](#)

Sobre desarrollo basado en Javascript puro, podés revisar unos videos de [Vanilla Javascript](#)

Expresiones regulares

Es bueno tener una idea sólida sobre expresiones regulares. Para arrancar gradualmente con ejercicios interactivos accedé a [RegexOne](#)

Para una revisión detallada podés optar entre:

- [Regular Expressions For Regular Folk](#)
- [Learn regular expressions in about 55 minutes - Sam Hughes](#)

Si preferís algo en video podés probar con [Regular Expressions Training Session](#)

Ansible

Ansible es una herramienta para el aprovisionamiento de servicios muy fácil de aprender y muy cómoda de utilizar basada en Python y YAML para aprovisionar fácilmente el software que debe ejecutarse en tu servidor web.

Para tener una idea del tema, dale un vistazo a esta playlist de [Ansible](#)