Curso de Web Scraping con Python

**Web Scraping**: proceso de extracción de datos [Objetivo: recopilar información]

**Web Crawling**: proceso de mapeo e indexación de páginas web para conocer su contenido diferente [Objetivo: conocer la estructura de la web]

**Scrapy:** es incluso usado en UK

¿Qué es un snipet?

**Ejemplos de Web Scrapeo**: descargar noticias diariamente, descargar información de competidores para hacer comparaciones.

IP (Internet Protocol: salen las direcciones iPv4 e iPv6 que identifican de manera única a las compus)

TCP (Transfer Control Protocol)

UDP (User Datagram Protocol)

TLS (Transport Layer Security [Encriptado])

DNS (Domain Name System [resuelve un nombre de dominio a un IP])

|  |  |  |
| --- | --- | --- |
| HTML | CSS | WEB APIs  JavaScript |
| HTTP | | |

**Inspector de elementos**: [presionar Tecla F12 en browser]

# Robots.txt

En Platzi.com/robots.txt (lo encontramos en Home)

User-Agent: \* (platzi dice que permite cualquier browser)  
Allow: / (que se puede visitar todo el sitio)  
Disallow: /\*/\*/concepto/\*/\*/material/ (no se puede entrar a esas partes de Platzi.com)  
Disallow: /\*/\*/live/ (por lo que veo, así evitas que el robot entre a los lives)

# XPath 101

$x('/') 🡪 El nodo raíz está representado con / además es un salto al siguiente nodo

$x('/html') 🡪 Se selecciona todo el documento

$x('//') 🡪 salto entre nodos, permite ir por varios niveles

$x('//h1') 🡪 regresa todos los h1

$x('//span/..') 🡪 consigue lo que está en el nodo padre

$x('//span/.') 🡪 consigue lo que está en el nodo actual

$x('//div/div/span/text()').map(x => x.wholeText) 🡪 consigue las citas, pero están bien desordenadas :s

$x('/html/body/div/div[1]') 🡪 trae el primer elemento div en el lugar indicado

$x('/html/body/div/div[last()]') 🡪 trae el ultimo

$x('//span[@class]') 🡪 trae todos los span que tengan, al menos, una class