Minería de Datos
IIC2433

Web Scraping

Vicente Domínguez

Datos

- ¿De dónde obtenemos los datos?
- En general pueden provenir de una base de datos, archivos o un *Data Warehouse*
- Y si no tengo nada de lo anterior, ¿qué hago?

La Web

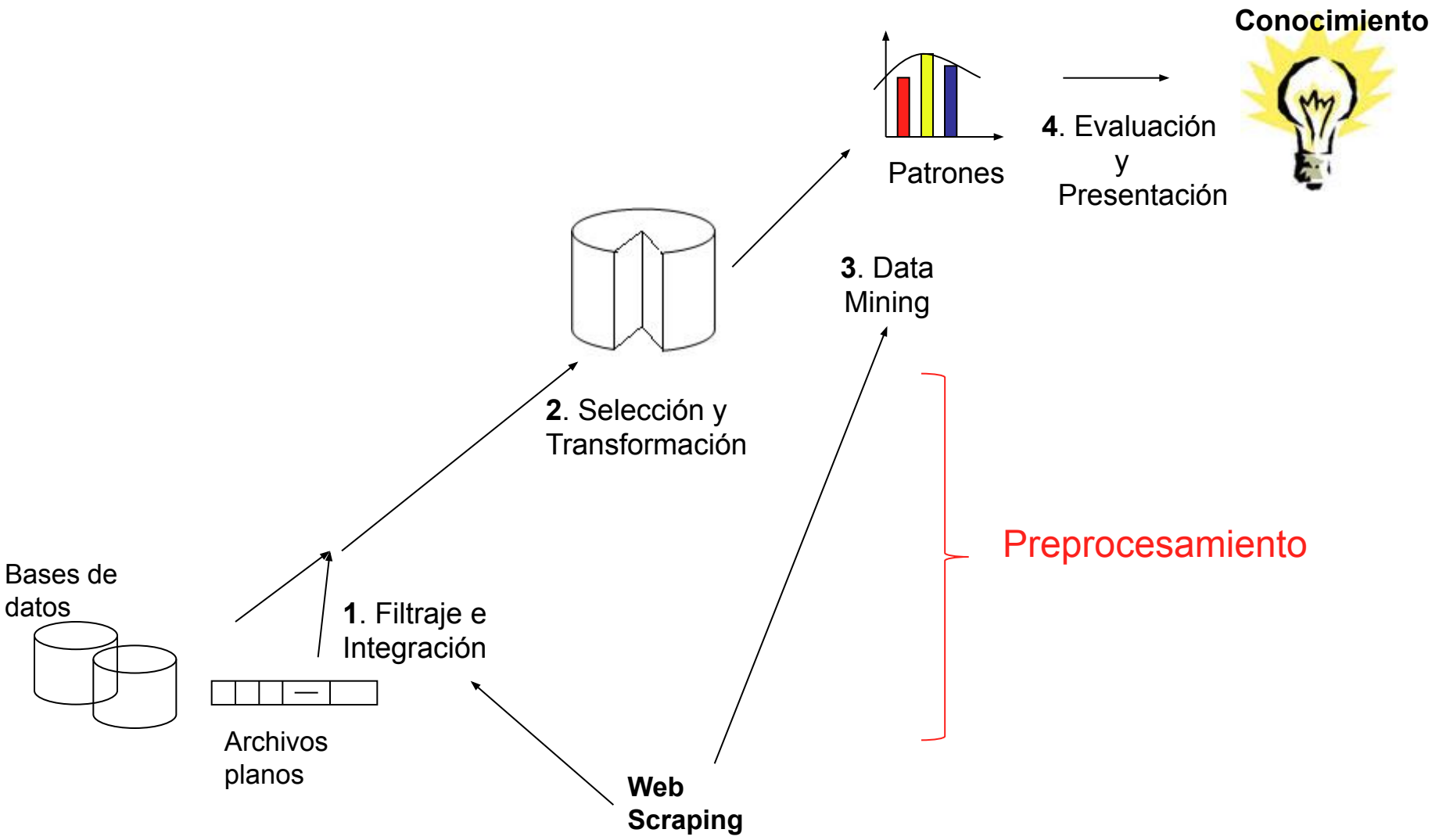
- La web está llena de datos (se podría decir que es la base de datos* más grande del mundo)
- Si no disponemos de una fuente directa de información, podemos obtenerla de acá.
- ¿Cómo obtenemos información estructurada de la web?

*** Base de datos de texto no estructurada**

Web Scraping

- Técnica para obtener información estructurada de la web.
- Requiere de algún lenguaje de programación o software que nos permita procesar la información.
- Podemos generar bases de datos de la información obtenida.

Knowledge Discovery in Databases



HTML

- El Lenguaje de Marcado de Hipertexto (HTML) es el utilizado para estructurar las páginas web.
- La mayoría de las páginas web están formadas por archivos en formato HTML.
- Si entendemos su estructura, entenderemos cómo obtener información de ellos.

```
<!DOCTYPE html>
<html>
    <head>
        <title>Example<title>
        <link rel="stylesheet" href="styl
    </head>
    <body>
        <h1>
            <a href="/">Header</a>
        </h1>
        <nav>
            <a href="one/">One</a>
            <a href="two/">Two</a>
            <a href="three/">Three</a>
        </nav>
```

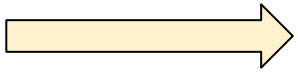
HTML

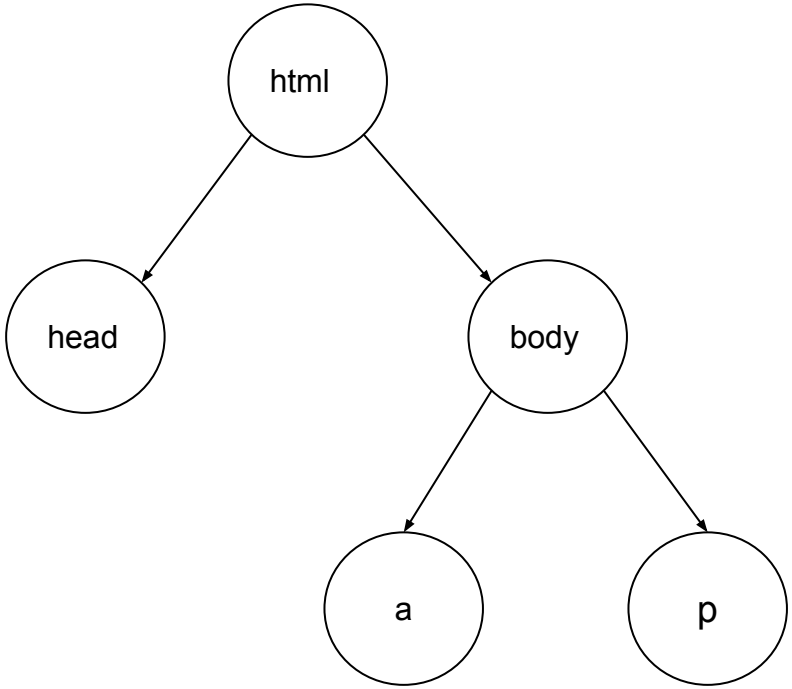
- Forma una estructura de árbol
- Los elementos se forman por tags de apertura y cierre
- Un elemento puede contener elementos dentro

```
<html>
    <head>
    </head>
    <body>
        <a></a>
        <p></p>
    </body>
</html>
```

HTML

```
<html>
    <head>
    </head>
    <body>
        <a></a>
        <p></p>
    </body>
</html>
```





HTML

- Cada elemento cumple funciones particulares.
- Para saber mucho más en detalle sobre HTML pueden revisar https://www.w3schools.com/html/
- A grandes rasgos hay 2 tipos de elementos.

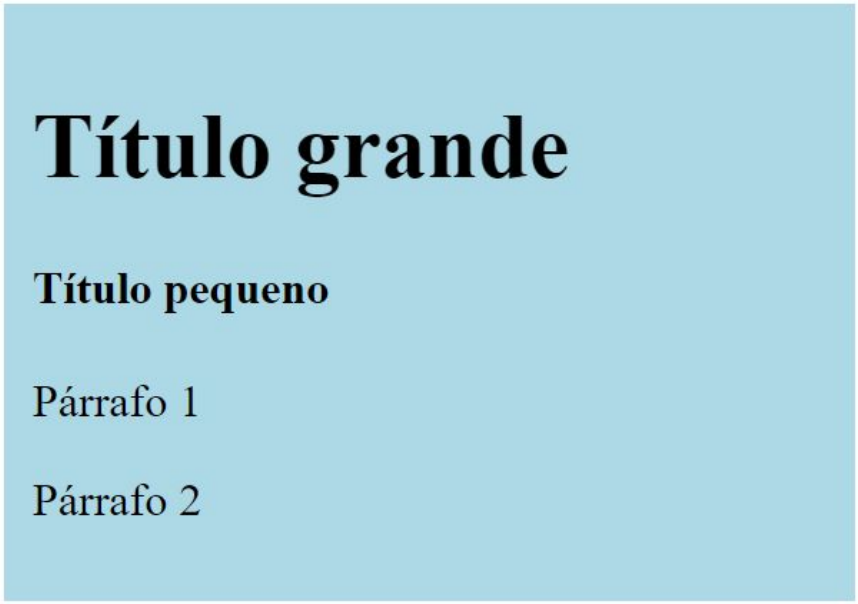
HTML- Elementos de Bloque

- Utilizan todo el ancho posible donde se ubican, aparecen en una nueva línea.
- Entre ellos se encuentran , <h1>...<h6>, <p>

```
<!DOCTYPE html>
<html>
<body>
<div style="background-
color:lightblue;padding:10px">
<h1>Título grande</h1>
<h4>Título pequeño</h4>

<p>Párrafo 1</p>
<p>Párrafo 2</p>

</div>
</body>
</html>
```



HTML- Elementos en línea

- No crean una nueva línea y solo utilizan el ancho necesario.
- Entre ellos , <img>, <a>

```
<!DOCTYPE html>
<html>
<body>
<div style="background-color:lightblue;padding:10px">
<h4>Título con <a href="https://www.ing.uc.cl/">enlace</a> </h4>
</div>
</body>
</html>
```

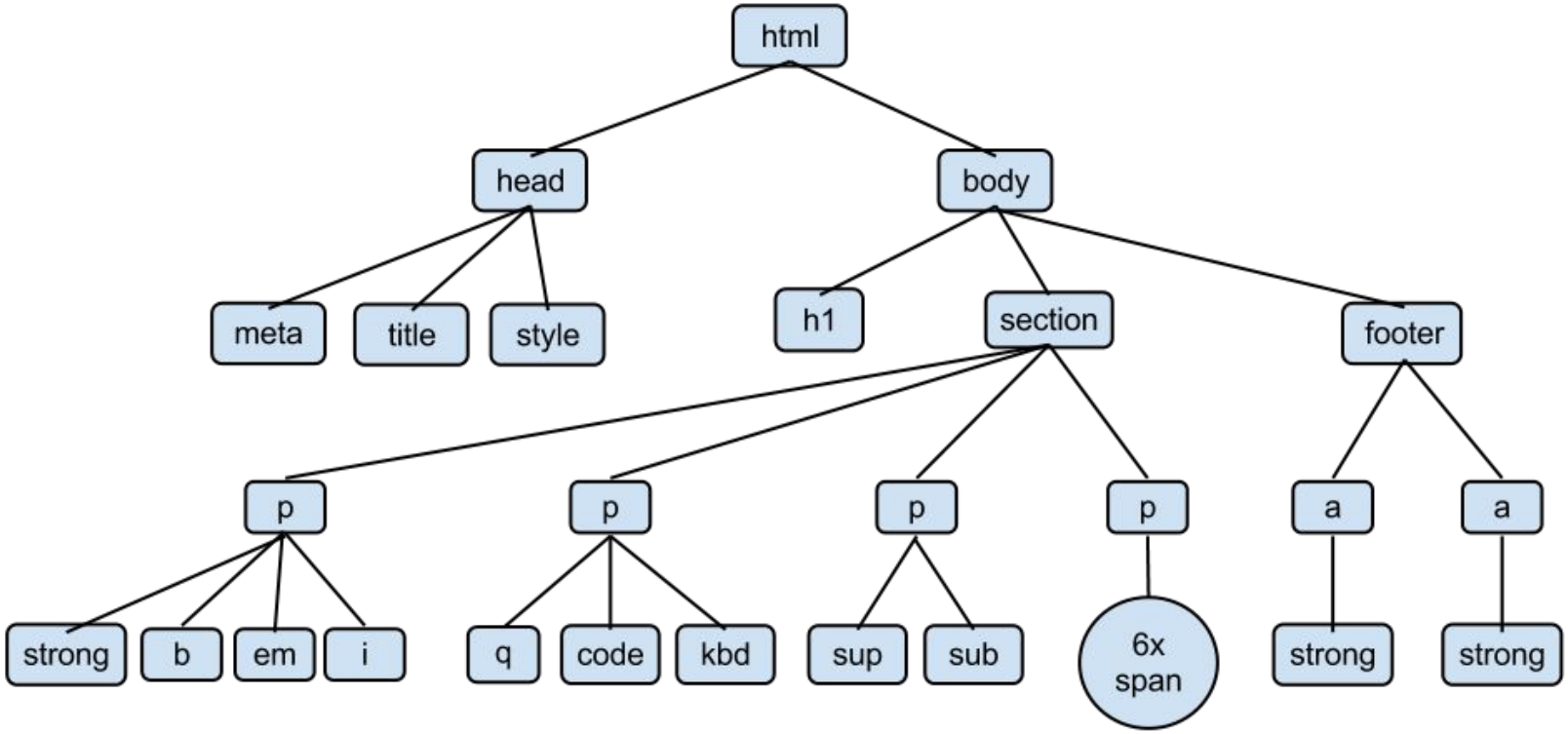
Web Scraping

- Hay una gran variedad de librerías y lenguajes para hacer web scraping.
- Utilizaremos Python y la librería Beautiful soup.
- Lo que se debe hacer es navegar la página como si fuera un **bot**, e ir almacenando la información.

Beautiful soup

- Nos permite transformar un documento HTML en un objeto de Python.
- Podemos recorrer y trabajar este documento como un **árbol**.
- Se puede consultar este árbol, y extraer información de él, consultando los **tags** HTML.

Beautiful soup



Fuente: https://medium.com/engineering-semantics3/web-scraping-in-perl-using-mojo-dom-a453229c732f

Beautiful soup

- Podemos acceder a los atributos y elementos del árbol como objetos, ej **`elemento.atributo`**
- Podemos buscar en el árbol todos los elementos de un tipo con los comandos **`arbol.find(elemento)`** y **`arbol.find_all(elemento)`**

Beautiful soup

- Podemos acceder a los atributos y elementos del árbol como objetos, ej **`elemento.atributo`**
- Podemos buscar en el árbol todos los elementos de un tipo con los comandos **`arbol.find(elemento)`** y **`arbol.find_all(elemento)`**

Actividad

- “¡Aún no sabemos usar Beautiful Soup!”
  - En la actividad aprenderemos a usarlo.

- Link de la actividad
  - La veremos paso a paso, y les explicaremos la dinámica de la actividad con Discord

- Link del servidor de Discord