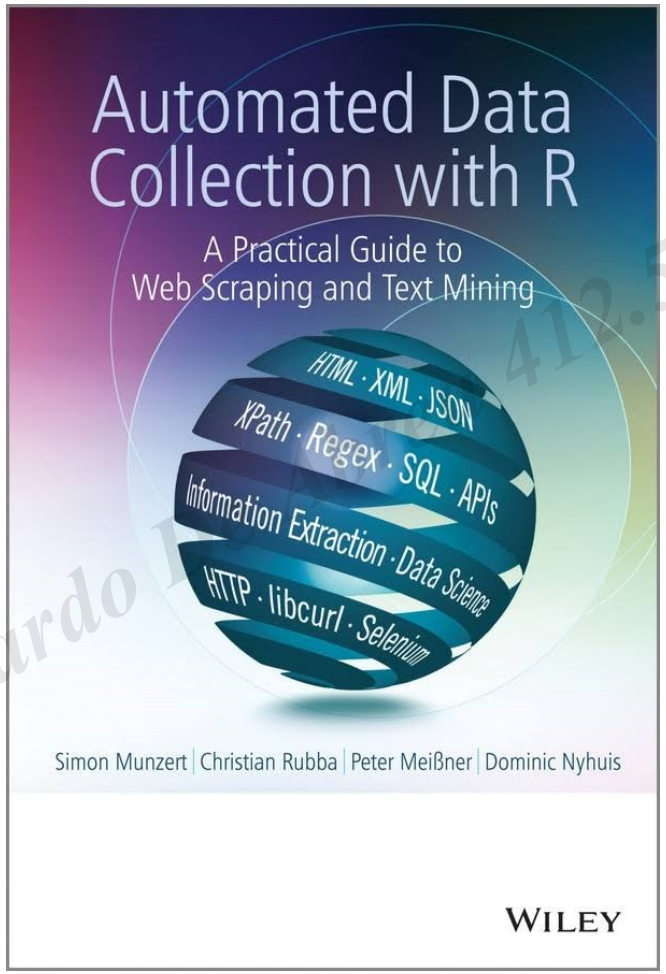
MBA
USP
ESALQ

COLETA DE DADOS: CRAWLERS E WEB SCRAPING

Prof. Dr. Jeronymo Marcondes

*A responsabilidade pela idoneidade, originalidade e licitude dos conteúdos didáticos apresentados é do professor.

Introdução



Introdução

- Plano de ataque:

1. Entendimento básico de como funciona a web.
2. Entendimento básico de HTML.
3. Entendimento básico de XML e JSON.
4. web scraping e o foco do curso.
5. Uso do R e de pacotes para realizar a operação.

Necessidade de Dados

- Muitos dados importantes = espalhados na web
- Locais mais comuns: webpages e APIs.
- Outras possibilidades: Servidores de FTP, por exemplo.
- Como recuperar dados da web e tratar a informação de forma a podermos utilizá-la e modelos e análises?

Estruturas de Dados

Os dados que podemos utilizar dividem-se em:

- Dados Estruturados.
- Dados Semiestruturados.
- Dados não estruturados.

Fonte: https://universidadedatecnologia.com.br/

- Estruturados - são os dados que detêm formatos bem definidos, como os extraídos de planilhas ou bancos de dados relacionais no formato SQL.
- Semiestruturados – Semelhantes aos dados estruturados, mas não obedientes na totalidade quanto à forma. Nesta linha estão os registros de linguagens baseadas em HTML e XML.
- Não estruturados ou NoSQL - não possuem um formato específico, são os dados coletados na sua forma original, como um texto, um vídeo, um fragmento de email, um log de sistema ou ainda uma mera foto.

Estruturas de Dados

- Como utilizar informação semiestruturada ou não estruturada que está na web, captá-la e armazená-la?

- WebScraping:

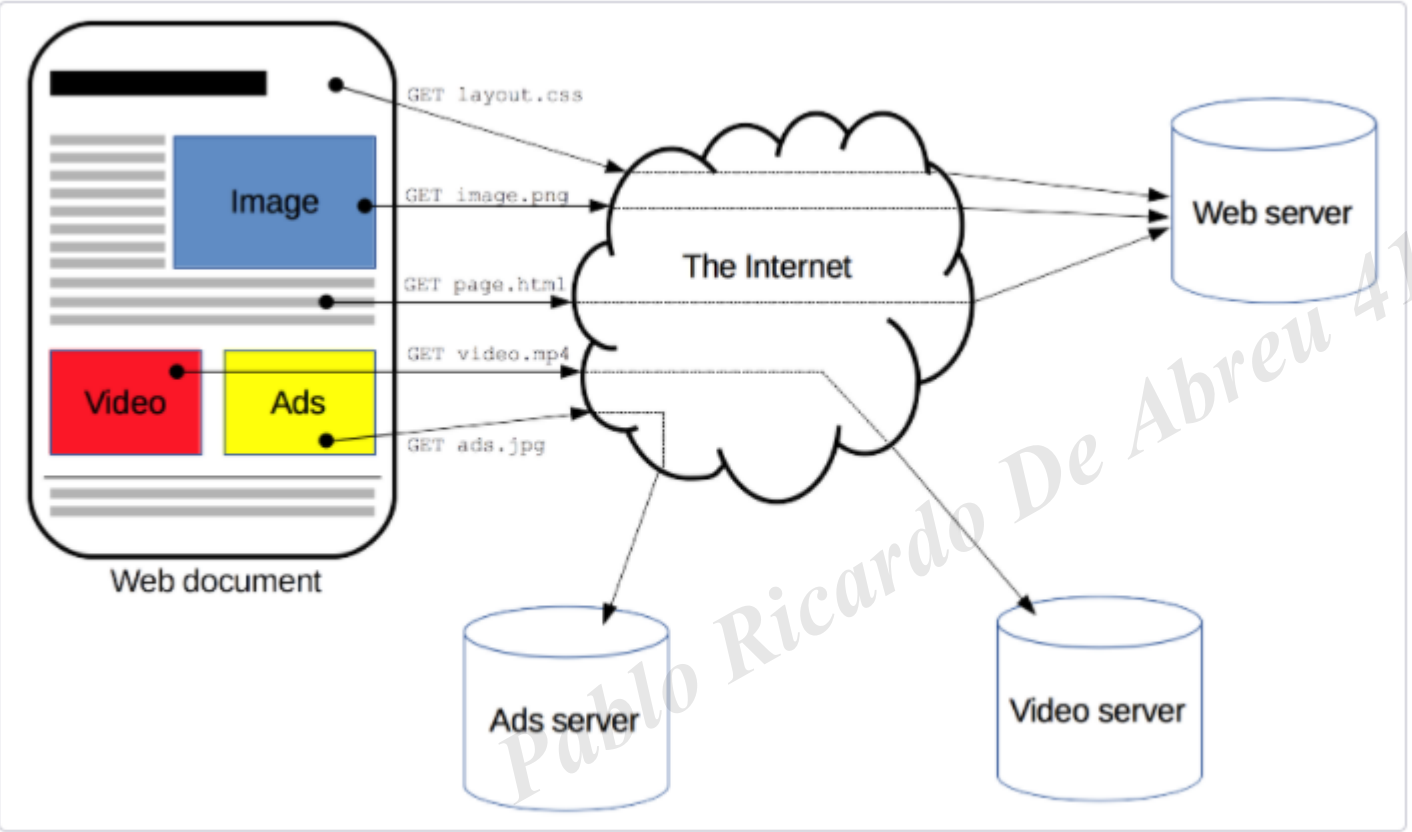
> O processo de extração e combinação de conteúdos de interesse da Web de uma forma sistemática. Em tal processo, um agente de software, também conhecido como robô, imita a interação de navegação entre os servidores da Web e o humano. Passo a passo, o robô acessa os sites conforme necessário, analisa seu conteúdo para encontrar e extrair dados de interesse e estruturar esses conteúdos conforme desejado (LOURENÇO, 2013).

Introdução à web

- HTTP

HTTP é um protocolo que permite a obtenção de recursos, como documentos HTML. É a base de qualquer troca de dados na Web e um protocolo cliente-servidor, o que significa que as requisições são iniciadas pelo destinatário, geralmente um navegador da Web.

Introdução à web



Fonte: https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Overview

Introdução à web

Clientes e servidores se comunicam trocando mensagens individuais. As mensagens enviadas pelo cliente, geralmente um navegador da Web, são chamadas de **solicitações** *(requests)*, ou também **requisições**, e as mensagens enviadas pelo servidor como resposta são chamadas de **respostas** *(responses)*.

Métodos de requisição

- Conjunto de tipos de requisição que indicam o que fazer
- HTTP verbs
- Não vamos estudar todos

Métodos de requisição

GET

O método GET solicita a representação de um recurso específico. Requisições utilizando o método GET devem retornar apenas dados.

POST

O método POST é utilizado para submeter uma entidade a um recurso específico, frequentemente causando uma mudança no estado do recurso ou efeitos colaterais no servidor.

Métodos de requisição

DELETE

O método DELETE remove um recurso específico.

CONNECT

O método CONNECT estabelece um túnel para o servidor identificado pelo recurso de destino.

HTML

- HTML
- HyperText Markup Language
- Linguagem posicional

HTML

http://www.r-datacollection.com/materials/html/OurFirstHTML.html

HTML

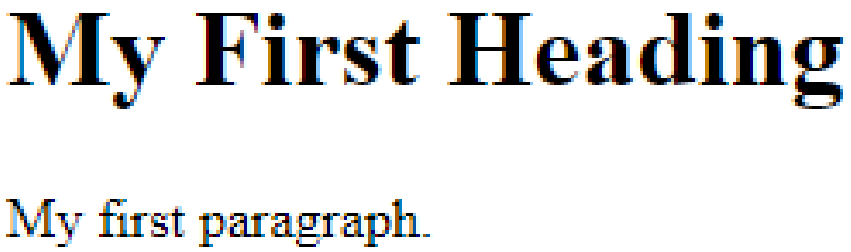
- Mais informações do que aparecem
- Etiquetas que mostram onde vai o conteúdo
- Explicam a estrutura do documento – sem layout

https://www.w3schools.com/html/html_basic.asp

HTML básico

Exemplo simples:

```
<!DOCTYPE html>
<html>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```



HTML - elementos

- Um elemento inicia com uma start tag <“nome da tag”> e finaliza com end tag <\”nome da tag”>

```
<tagname>Content goes here...</tagname>
```

- Tags aninhadas – exemplo anterior

HTML

- Doctype
- <html> e </html>
- <body> e </body> - todos elementos em uma página, como os headings, parágrafos, imagens, hiperlinks, etc

HTML

- Heading (h1, h2, h3, h4, h5 e h6)
- Parágrafos e

```
<!DOCTYPE html>
<html>
<body>

<p>Esse é um parágrafo.</p>
<p>Esse é outro parágrafo.</p>

</body>
</html>
```

Esse é um parágrafo.

Esse é outro parágrafo.

```
<!DOCTYPE html>
<html>
<body>

<h1>heading 1</h1>
<h2>heading 2</h2>
<h3>heading 3</h3>
<h4>heading 4</h4>
<h5>heading 5</h5>
<h6>heading 6</h6>

</body>
</html>
```

heading 1

heading 2

heading 3

heading 4

heading 5

heading 6

HTML

- Hiperlink: <a> e :

```
<!DOCTYPE html>
<html>
<body>

<h1>Aqui está o heading</h1>

<a href="https://mbauspesalq.com/">Data Science</a>

</body>
</html>
```

Aqui está o heading

Data Science

HTML

- div: e : define uma divisão ou seção de uma página HTML
- Container para um conjunto de tags HTML – características específicas de uma parte específica da página.

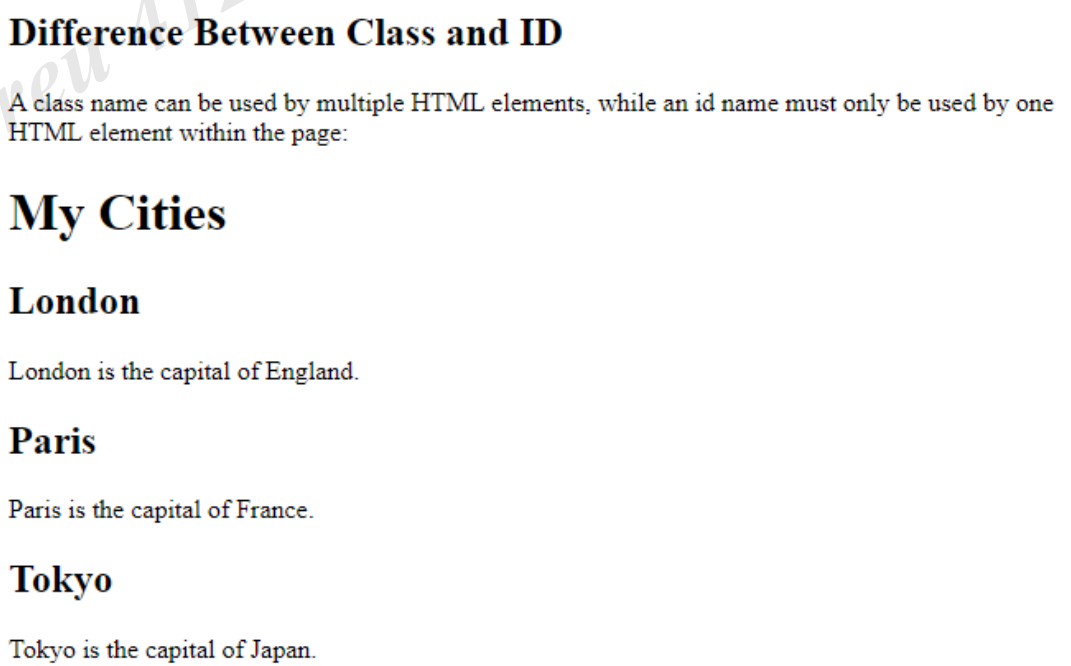
Fonte: https://www.geeksforgeeks.org/

HTML

- Atributos: informação adicional sobre os elementos
- Id e Class

HTML

```
<!DOCTYPE html>
<html>
<body>

<h2>Difference Between Class and ID</h2>
<p>A class name can be used by multiple HTML elements, while an id name must
only be used by one HTML element within the page:</p>

<!-- An element with a unique id -->
<h1 id="myHeader">My Cities</h1>

<!-- Multiple elements with same class -->
<h2 class="city">London</h2>
<p>London is the capital of England.</p>

<h2 class="city">Paris</h2>
<p>Paris is the capital of France.</p>

<h2 class="city">Tokyo</h2>
<p>Tokyo is the capital of Japan.</p>

</body>
</html>
```



API

- Novas formas de compartilhar dados na web
- Crescente o uso de APIs: Application Programming Interface
- Conjunto de definições, protocolos e regras para criar integrações entre sistemas.

API

- Como ela funciona:

1. Aplicação cliente envia requisição para API
2. API verifica se a requisição é válida
3. Se for válida processa o necessário para resposta
4. A API transfere os dados para a aplicação cliente

- Exemplo: IBGE

API

- Arquitetura REST

- Mesmos verbos do HTTP – utiliza as características do mesmo, como por exemplo:

1. Protocolo cliente-servidor
2. Mensagens HTTP

API

- Vantagens para as empresas – microserviços
- Retorno dos dados: maior parte das vezes JSON
- Mas, tipo de retorno pode ser variado

XML

- Uma das formas mais populares de trafegar dados na web
- Mesma base markup do HTML, mas ao invés de mostrar informação, o objetivo é guardar dados.
- Dados guardado em um texto puro – qualquer navegador ou sistema operacional consegue entender
- Tags auto descritivas

XML

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<bond_movies>
  <movie id="1">
    <name>Dr. No</name>
    <year>1962</year>
    <actors bond="Sean Connery" villain="Joseph Wiseman"/>
    <budget>1.1M</budget>
    <boxoffice>59.5M</boxoffice>
  </movie>
  <movie id="2">
    <name>Live and Let Die</name>
    <year>1973</year>
    <actors bond="Roger Moore" villain="Yaphet Kotto"/>
    <budget>7M</budget>
    <boxoffice>126.4M</boxoffice>
  </movie>
  <movie id="3">
    <name>Skyfall</name>
    <year>2012</year>
    <actors bond="Daniel Craig" villain="Javier Bardem"/>
    <budget>175M</budget>
    <boxoffice>1108.6M</boxoffice>
  </movie>
</bond_movies>
```

XML

- Características:

1. Versão do XML
2. Tags de início e fim
3. Elementos
4. Atributos
5. Dados
6. Estrutura em árvore – nós pais e filhos

XML

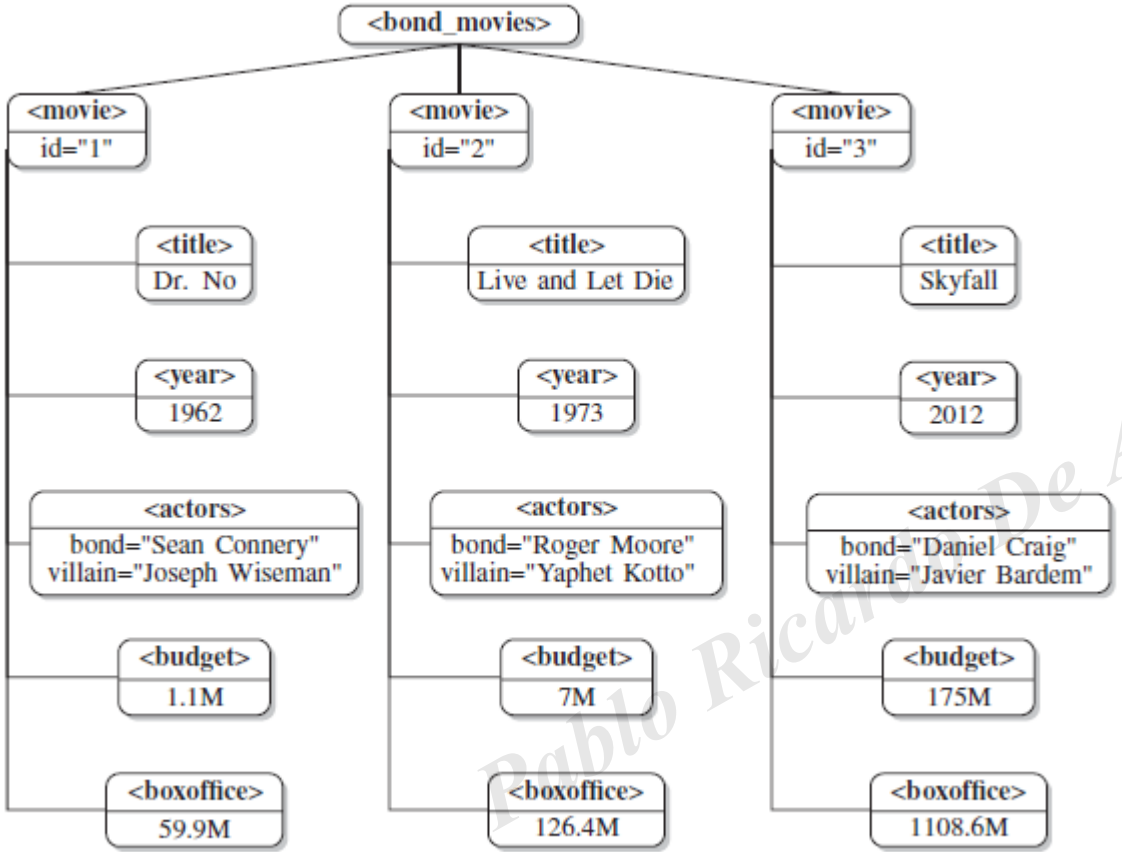


Figure 3.2 Tree perspective on an XML document

JSON

```
{"indy movies" :[
      {
      "name" : "Raiders of the Lost Ark",
      "year" : 1981,
      "actors" : {
            "Indiana Jones": "Harrison Ford",
            "Dr. René Belloq": "Paul Freeman"
            },
      "producers": ["Frank Marshall", "George Lucas", "Howard Kazanjian"],
      "budget" : 18000000,
      "academy_award_ve": true
      },
      {
      "name" : "Indiana Jones and the Temple of Doom",
      "year" : 1984,
      "actors" : {
            "Indiana Jones": "Harrison Ford",
            "Mola Ram": "Amish Puri"
            },
      "producers": ["Robert Watts"],
      "budget" : 28170000,
      "academy_award_ve": true
      },
      {
      "name" : "Indiana Jones and the Last Crusade",
      "year" : 1989,
      "actors" : {
            "Indiana Jones": "Harrison Ford",
            "Walter Donovan": "Julian Glover"
            },
      "producers": ["Robert Watts", "George Lucas"],
      "budget" : 48000000,
      "academy_award_ve": false
      }]
}
```

JSON

- Características:

1. Baseada em objetos (semelhante aos elementos do XML)
2. Baseada na divisão de pares key\value
3. Keys estão sob aspas duplas
4. Valores são precedidos de dois pontos
5. Valores estarão sob aspas duplas se forem campos de texto
6. Array são definidos por colchetes

Web Scraping

- Como buscar a informação?
- Pacotes que executam os verbos em HTTP, como GET
- Pacotes httr e rvest

Web Scraping

- Retorno como texto
- PARSE
- Busca a informação em determinado ponto

Web Scraping

Discussão: web scraping é ilegal?

OBRIGADO!

linkedin.com/in/jeronymo-marcondes-585a26186

40