PES UNIVERSITY

Web Scraping

Dr.Mamatha.H.R

Professor

Department of Computer Science and Engineering

PES University

Bangalore

Click to add Text

Course material created using various Internet resources

What is web scraping

- Web scraping is used to extract or “scrape” data from any web page on the Internet.
- Copying a list of contacts from a web directory is an example of “web scraping”.

- Web scraping is performed using a **“web scraper”** or a “bot” or a “web spider” or “web crawler” .
- A web-scraper is a program that goes to web pages, downloads the contents, extracts data out of the contents and then saves the data to a file or a database

How vital is Web Scraping

- The Internet would be far less useful and terribly small without Web Scraping.
- The lack of availability of “real integration” through APIs has turned Web Scraping into a massive industry with trillions of dollars in impact on the Internet economy.
- Google
- McKinsey put a number of 8 trillion dollars on it in 2011 and it has only increased exponentially since.
- *There is an enormous amount of data “available” on the Internet but it is hardly “accessible”.*
- **Web scraping makes this data accessible** to all kinds of applications and uses

Web Crawling vs. Web Scraping

- Web Crawling mostly refers to downloading and storing the contents of a large number of websites, by following links in web pages.
- Search Engines depend heavily on web crawlers.
- Googlebot is an example of a web crawler.

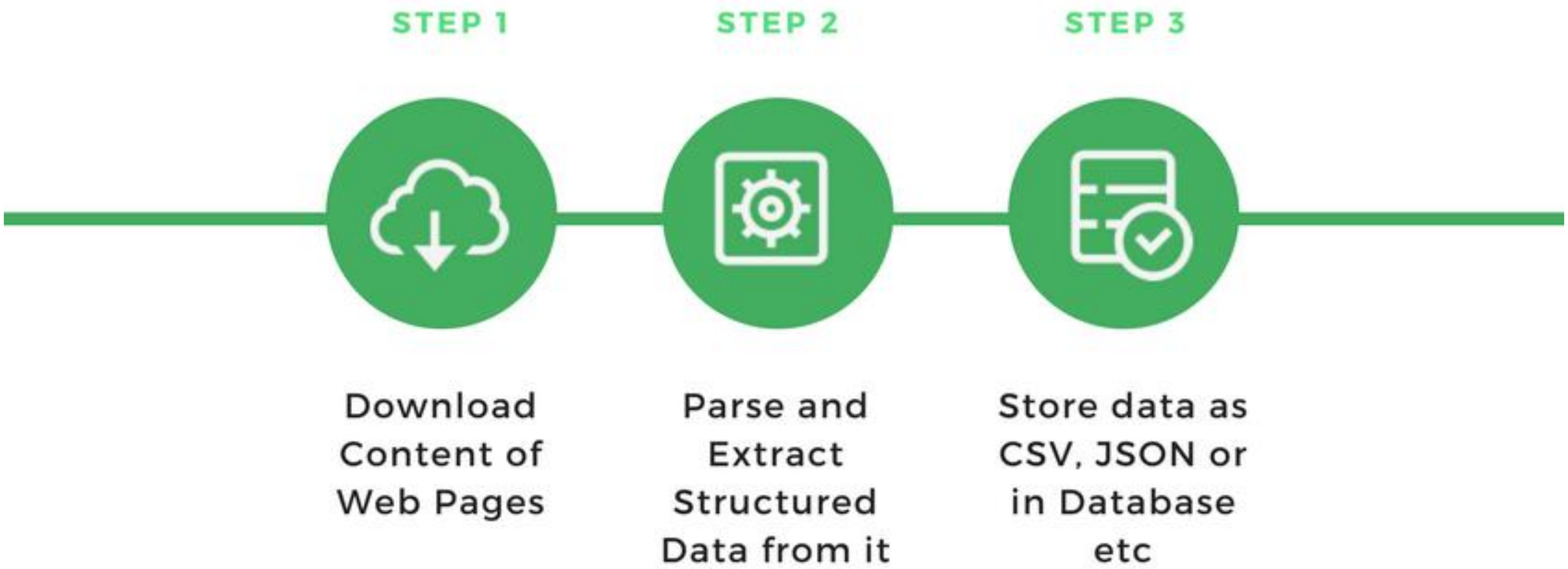
- A Web scraper is built specifically to handle the structure of a particular website.
- The scraper then uses this site-specific structure to extract individual data elements from the website.
- The data elements could be names, addresses, prices, images etc.
- For example, SERP monitoring services scrape search engine results periodically to show you how your search rankings have changed over time. They use a separate scraper for each search engine.

Uses of Web Scraping

- Search Engines
- Price Monitoring
- Sales and Marketing
- Content Aggregators
- Sales intelligence
- Training datasets for Machine Learning
- Data for Research

How does a web scraper work?

- A web scraper is a software program or script that is used to download the contents (usually text based and formatted as HTML) of multiple web pages and then extract data from it.



STEP 1

Download Content of Web Pages

STEP 2

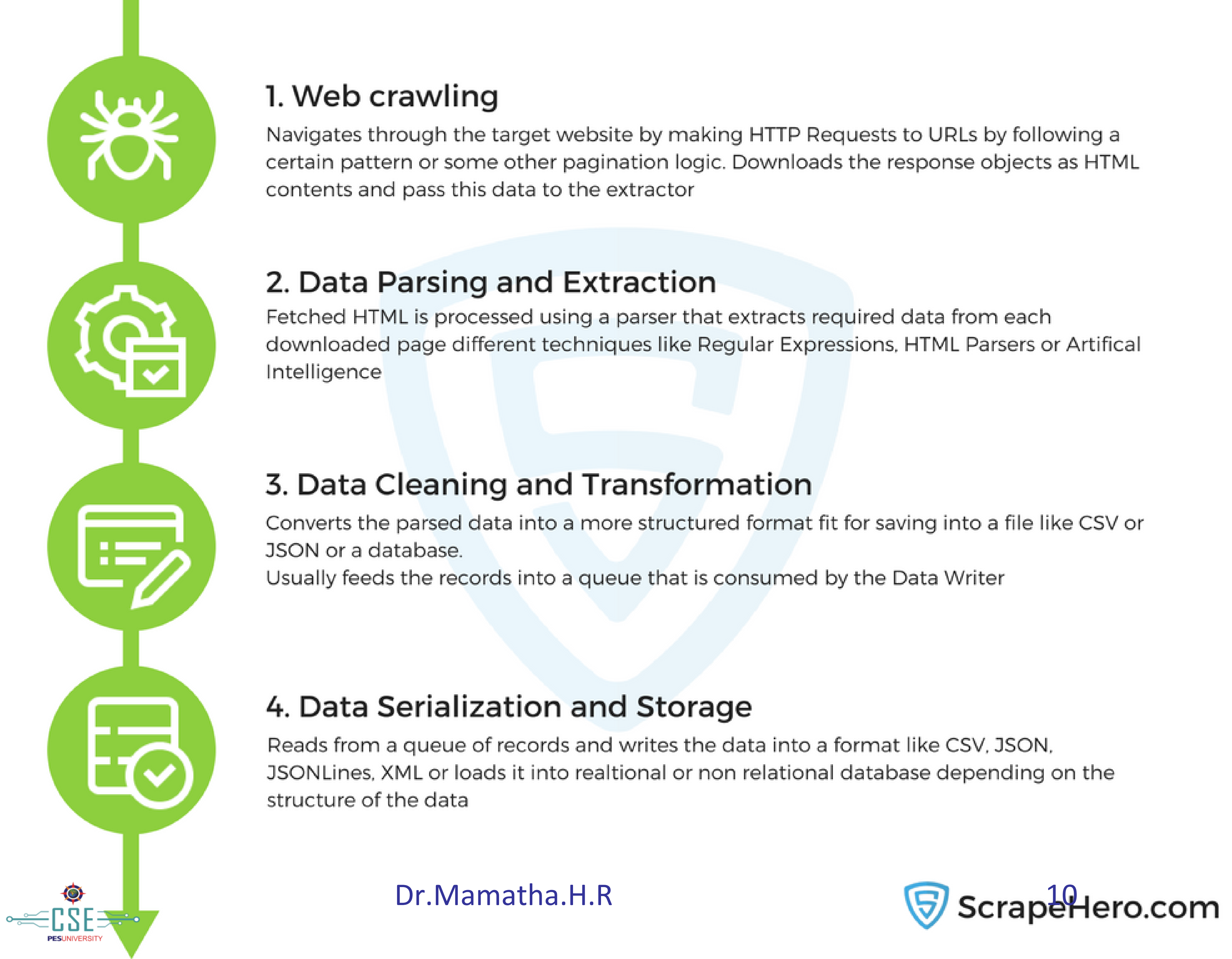
Parse and Extract Structured Data from it

STEP 3

Store data as CSV, JSON or in Database etc

What are the components of a web scraper

- Web scraping is like any other Extract-Transform-Load (ETL) Process.
- Web Scrapers crawl websites, extracts data from it, transforms to a usable structured format and load it to a file or database for subsequent use.



1. Web crawling

Navigates through the target website by making HTTP Requests to URLs by following a certain pattern or some other pagination logic. Downloads the response objects as HTML contents and pass this data to the extractor

2. Data Parsing and Extraction

Fetched HTML is processed using a parser that extracts required data from each downloaded page different techniques like Regular Expressions, HTML Parsers or Artifical Intelligence

3. Data Cleaning and Transformation

Converts the parsed data into a more structured format fit for saving into a file like CSV or JSON or a database.

Usually feeds the records into a queue that is consumed by the Data Writer

4. Data Serialization and Storage

Reads from a queue of records and writes the data into a format like CSV, JSON, JSONLines, XML or loads it into realtional or non relational database depending on the structure of the data

What is the best Programming Language to build a web scraper?

- You can build web scrapers in almost any programming language.
- It is easier with Scripting languages such as Javascript (Node.js), PHP, Perl, Ruby or Python.
- Irrespective of the programming language, you will need some libraries to
- Make HTTP Requests, receive Responses and get the HTML content
- A Data Parser
- A Data Serializer

- Python has an excellent library called Requests (built on top of another similar library urllib2) for downloading web pages,
- libraries such as BeautifulSoup and LXML for parsing the HTML.
- libraries such as Pandas, MatPlotLib, Numpy, Scipy, etc makes python an excellent choice for handling, transforming and using data for various applications.

- Send a request to https://abc.com/ and download the HTML Content of the page.
- as we are only extracting data from a single link this scraper does not need a web crawling component .
- Python's built-in URL handling library – urllib to download the HTML of the web page.
- Parse the downloaded data using an HTML Parser to extract some data. (The scraper's parser module).
- For parsing the HTML, we will use BeautifulSoup 4, a library used for pulling data out of HTML and XML files. It works with HTML parsers to provide idiomatic ways of navigating, searching and modifying the parse tree.

- Transform the data into a usable format
- Print the extracted data into the terminal (or console) and also save the data to a respective file