03 | Databases, Scrapping, and APIs

Ivan Corneillet

Data Scientist



Learning Objectives

After this lesson, you should be able to:

- Access SQL databases from pandas
- Describe how web scraping works, conceptually and explain practically how to web scraping works using Python
- Define how to approach scraping project data
- Describe APIs and how to make calls and consume API data

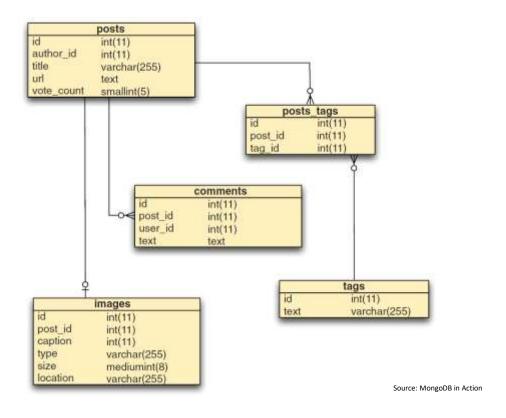
Here's what's happening today:

- RDBMS/SQL Databases
 - The SELECT statement
- Scrapping
- Application Programming Interfaces (APIs)



Relational Database Management Systems (RDBMS) and Structured Query Language (SQL)

A relational database links data entities and concepts. E.g., a relational data model for entries on a social news site



Relational databases are organized into tables. Each table has a specific schema, a set of rules for what goes in each table

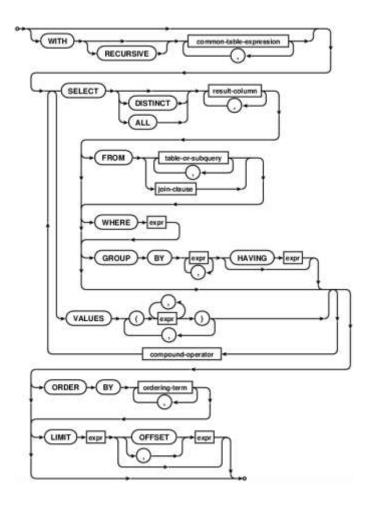
- Each table corresponding to one entity or concept
- A table is made up of rows and columns, similar to a *pandas* dataframe
- Schemas specify which columns are contained in the table and what type of data is in each column (e.g., text, integer, or date). This means you can't add text data to an integer column
- For this reason and many others,
 databases allow for stronger consistency
 of the data and are often a better solution
 for data storage

CRUD and SQL

CRUD ^(*) (*) the four basic functions of persistent storage	SQL Statement	
Create	CREATE	Create or add new entries
Read	SELECT	Read, retrieve, search, or view existing entries
Update	UPDATE	Update or edit existing entries
Delete	DELETE	Delete/deactivate/remove existing entries

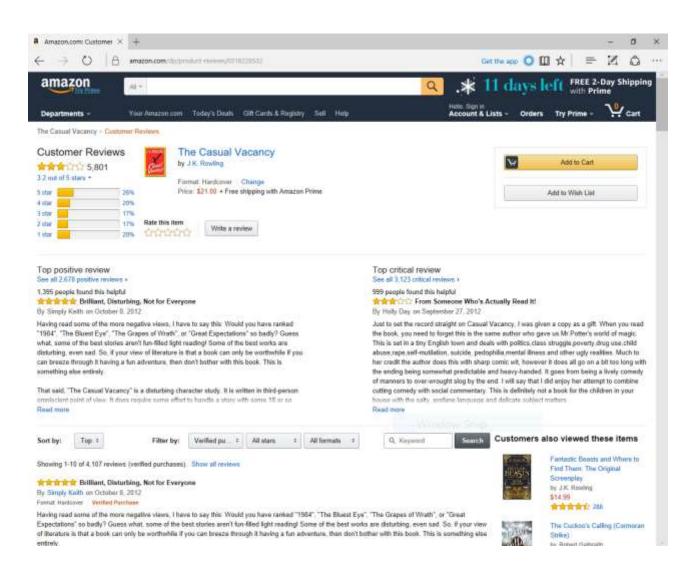
SQLite's **SELECT** Statement

(https://www.sqlite.org/lang_select.html)

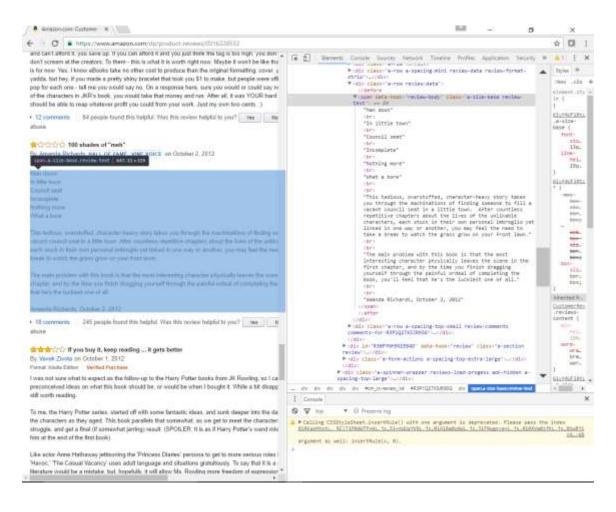




Scraping

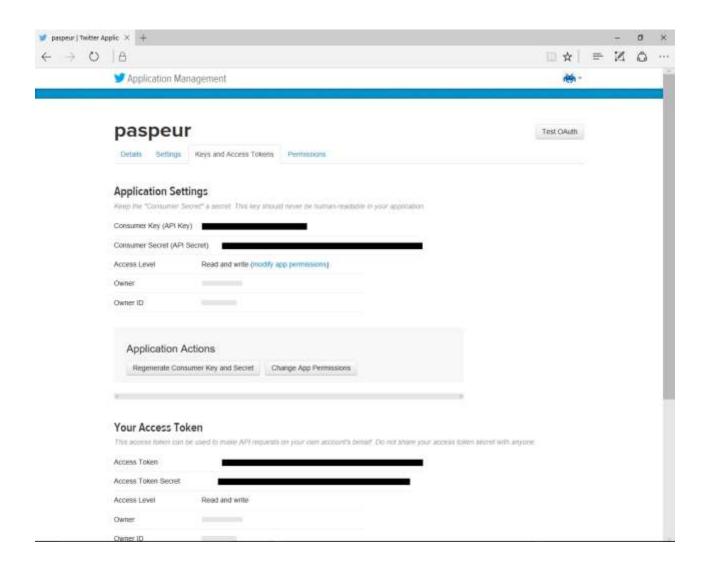


Google Chrome's inspect element feature is a great tool to scrape web sites





Application Programming Interfaces (APIs)



Slides © 2017 Ivan Corneillet Where Applicable Do Not Reproduce Without Permission