The Powerball Summary

By Mojalefa Thapo

Introduction

While predicting winning Powerball numbers remains elusive, this project explores how close we can get to approximating their distribution. We'll leverage descriptive statistics and data analysis techniques to develop a tool that estimates the Powerball number landscape. Additionally, we'll build a database for easy querying of historical results. It's important to acknowledge that due to the inherent randomness of the lottery, a truly predictive model based solely on past data is not achievable.

Project Description

The proposed system has been designed to scrape powerball results from the web and perform a descriptive analysis on the obtained data. The scraper utilises BeautifulSoup to fetch the HTML elements containing the dates and winning numbers. The project leverages two Python files to automate the data extraction and processing workflow. The extracted data is then cleaned and formatted for further analysis using Pandas and excel to create a basic relational database using MS Access. This will be achieved through the implementation of a streamlined ETL system and the utilisation of descriptive statistics and data visualisation techniques while facilitating data querying capabilities.

The ultimate goal is to provide powerball players with an overall structure of the numbers drawn between 2016 and 2023. It is crucial to emphasise that the generated insights will constitute an approximation and should not be misconstrued as guaranteed predictions of future Powerball outcomes.

The excel file containing the descriptive analysis and python files can be obtained on my public github repository on https://github.com/Tyler-Uchiha/Scrape-and-clean-PB/tree/main

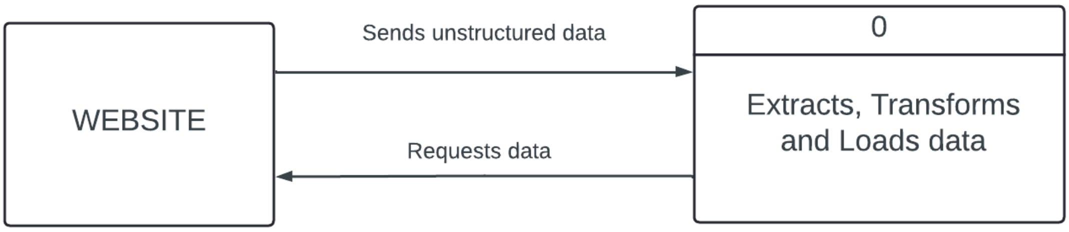
How data flows

The ETL system's data flow utilises a two-tiered approach based on Gane and Sarson notation.

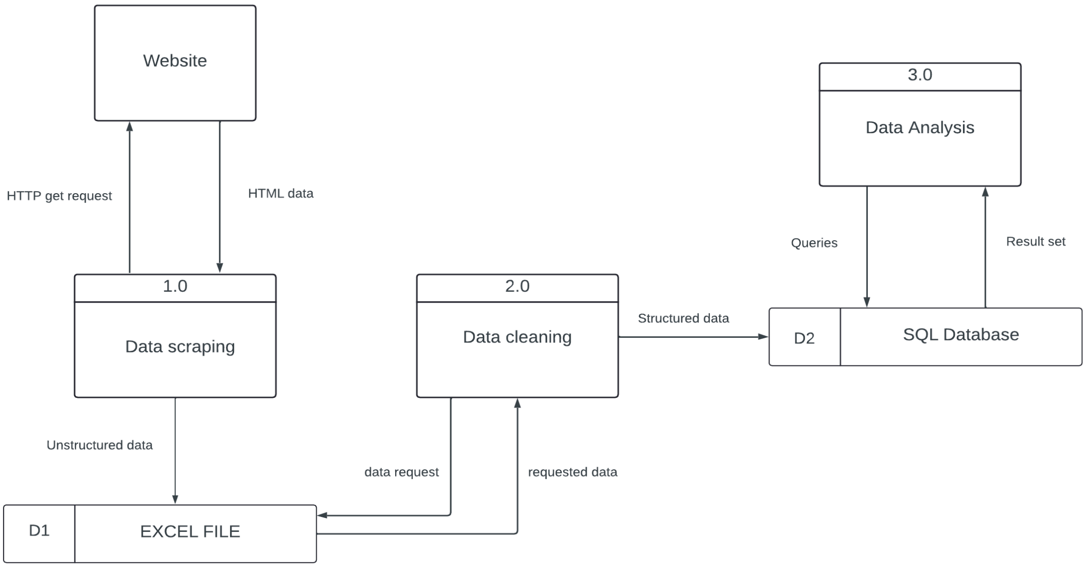
Level 0 (Context Diagram): This high-level view depicts the entire ETL system interacting with a single external entity – the data source website. It focuses on the overall flow of data without delving into technical details.

Level 1 (Detailed Diagram): This level dives deeper, illustrating the specific processes within the ETL system for data extraction, transformation, and loading in relation to the website source. It clarifies how the ETL system accomplishes its core functions.

DFD LEVEL 0



DFD LEVEL 1



The ETL process unfolds in a series of well-defined steps:

Data Extraction: The journey begins with a web scraper. This script, often utilising libraries like BeautifulSoup (bs4), sends an HTTP GET request to the target website (data source). The website responds by delivering the requested HTML data. This raw, unstructured data becomes the initial haul. The scraper meticulously stores this data within an Excel file, acting as our first data store.

Data Transformation: The baton is passed to a separate Python script. This script leverages the power of pandas to access and cleanse the data residing in the Excel file. The desired data is meticulously extracted and fed back into the transformation process for further manipulation. After undergoing transformations to ensure accuracy and consistency, the data is readied for its final destination.

Data Loading: The transformed data embarks on its final leg. It's meticulously loaded into the second data store, which in this instance is an SQL database. This database serves as a structured and organised repository for the valuable insights gleaned from the website.

Data Analysis: Analysts can now leverage SQL queries to retrieve specific data points from the database. This retrieved data, often referred to as a result set, becomes the foundation for further exploration and visualisation. Through these visualisations, valuable trends and patterns can be unearthed, providing actionable insights.

Project Structure

The project consists of two python files:

1. scraper.py

2. scraper_and_cleaner.py

3. SQL Database

To run the files

1. Run scraper.py to scrape the Powerball results and generate an initial Excel file.

2. Run scraper_cleaner.py twice with the following arguments:

   First run: you need to pass "data" as an argument into pd.DataFrame() and rename the outputted excel file accordingly, to output powerball numbers (This creates an Excel file containing the cleaned Powerball numbers)

   Second run: you will need to pass "dates" as an argument into pd.DataFrame() to generate the dates of all the draws.