|  |
| --- |
| Florida Lottery Analysis |
| Team: Martie Feldtmann, Lisette Cruz |

# Project Report

## Project Proposal

After some initial discussions we agreed on using the Lottery results, mainly focusing on Florida, to perform the Extract, Transform and Load activities. The initial expectation was to mainly use the Florida Lottery website as out source of data and break the tasks up in the type of extract that needed to be performed, namely, PDF extracts and HTML extracts.

The table below formed part of our initial discussion around the subject of the project:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Game | Datasets | Dataset1 Table | Dataset2 Table | Joined Table |
| Florida Lottery | * Winning Numbers History – HTML * Payouts (2019) – PDF | * Date * 6 Numbers * Multiplier | * Date * Total Sales * Rollover (Boolean) * Jackpot (if Rollover = False) * Total winners | * Date * 6 Numbers * Multiplier * Total Sales * Rollover * Jackpot |
| Powerball | * Winning numbers – HTML * Jackpot winners – PDF | * Date * 5 Numbers * Powerball * Multiplier | * Date * Winning City * Retailer name * Advertised Jackpot * Actual Jackpot * Prize Payout * Jackpot winning tickets | * Date (All rows from Winning numbers which will indicate which were rollovers) * 5 Balls * Powerball * Multiplier * Winning city * Retailer name * Advertised Jackpot * Actual Jackpot * Prize Payout * Jackpot winning tickets |
| Jackpot Triple Play | * Payouts (2019) – PDF | * Date * 6 Numbers * Total Sales * Prize Pool |  |  |
| Fantasy 5 | * Winning Numbers History – PDF | * Date * 6 Numbers * Multiplier |  |  |

After the extracts, the data would be cleaned and reformatted before using pandas dataframes to write the results to the MongoDB database. The data we extracted pretty much fitted well for using the relational database.

## Discovery

The discovery phase of the project included looking more carefully at the data sources available on the Florida Lottery website. There is a large number of data for multiple games available. Some of the data is available in HTML format, while the majority is stored as pdf documents. As we have not worked with pdf documents before, it seemed like a nice challenge to attempt the data extract from these types of files. Both the HTML and PDF data can be read directly from the website, so we did not need to download the files and store it locally first.

Although the HTML did not proof to be as straight forward as expected, we could still use Beautifulsoup to assist with the extract.

As we have not used any tools to extract data from pdf during the class exercises, we had to investigate various possible ways to extract the data out of these files. We played around with a couple of python dictionaries to see what the results look like. These dictionaries included :

* Tabula
* PDFPlumber
* PDFMiner
* Tika

We did however decide to stick with Tika as it looked like the cleanest extract.

We further decided to also include other websites to extract data for the Powerball. These sites included Lottery Corner (HTML data) and the NY Government site (XML data).

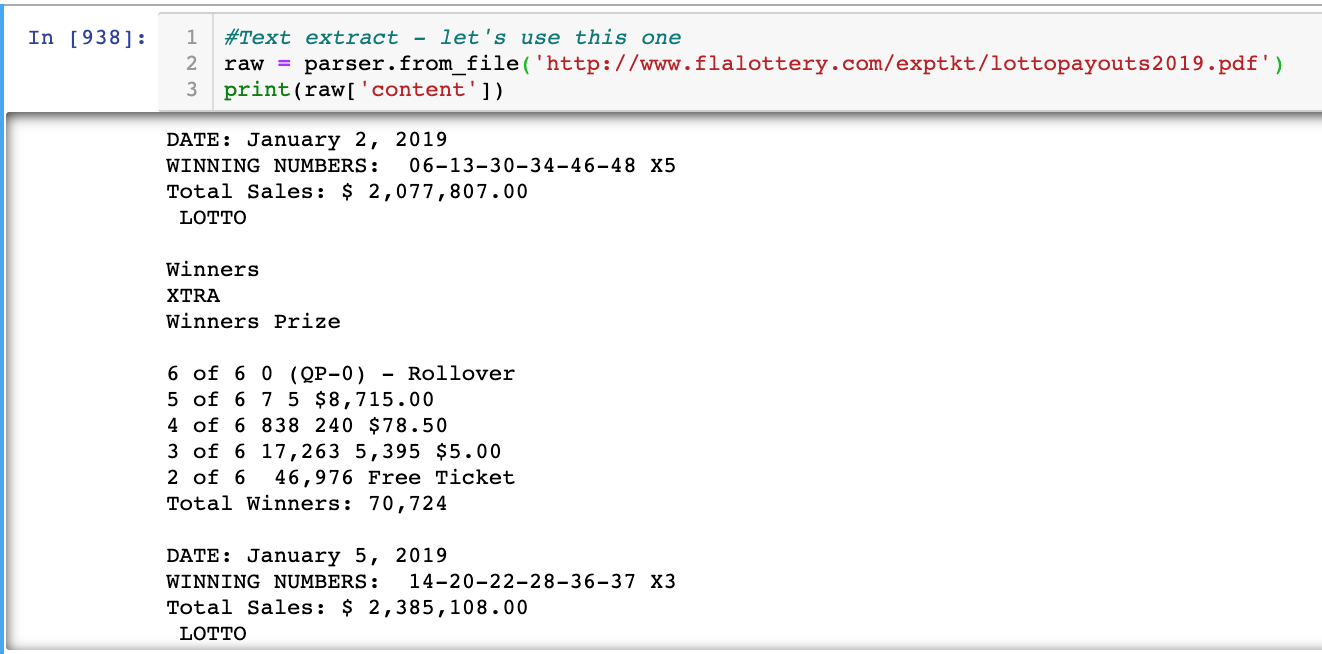
## Extract

The following table illustrate the different data sources :

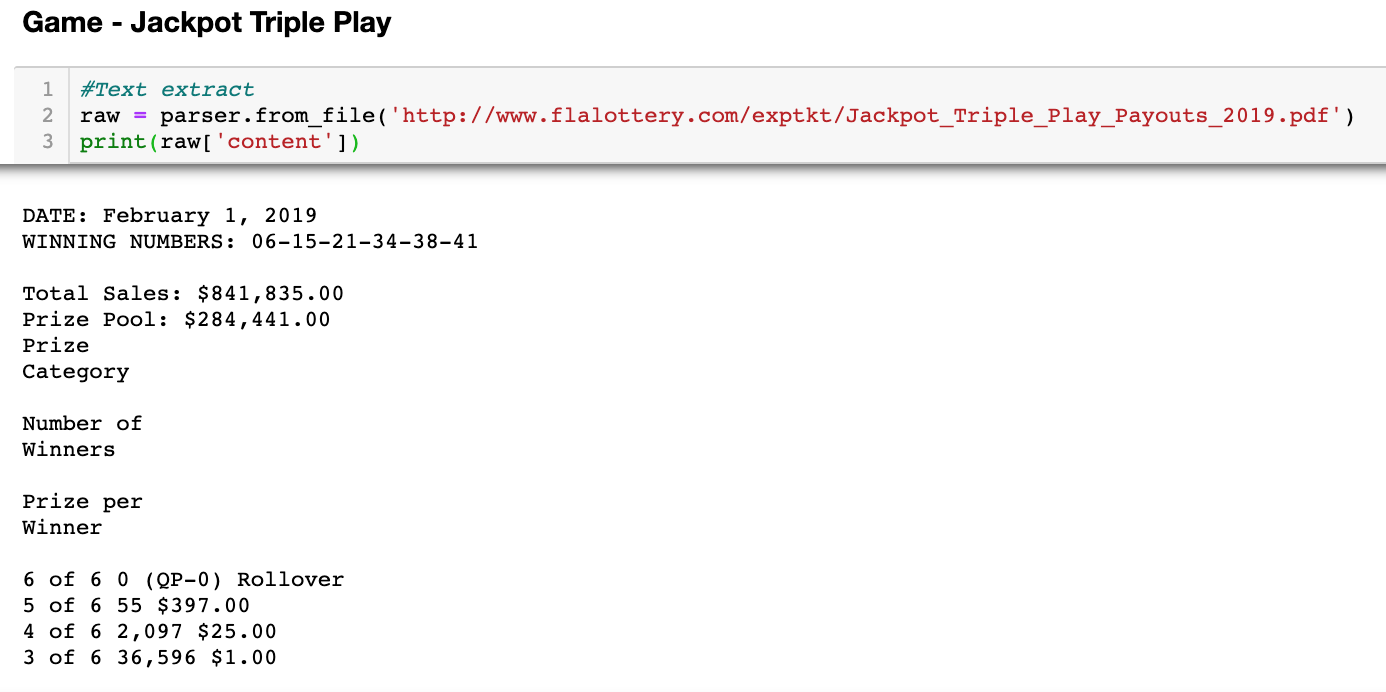
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Website | Game | Data Source Type | Content | Fields |
| <http://flalottery.com/reports> | Florida Lotto | PDF | 2019 payouts | * Date * Total Sales * Winning Numbers * Multiplier * Total Winners |
| Jackpot Triple Play | PDF | 2019 payouts | * Date * Total Sales * Winning Numbers * Prize Pool |
| Fantasy 5 | PDF | 2019 payouts | * Date * Total Sales * Winning Numbers |
| Powerball | PDF | Lotto Winners Dec 09 to present | * Date * Advertized Jackpot * Prize Payout * Number of Winners |
| <https://catalog.data.gov/dataset/lottery-powerball-winning-numbers-beginning-2010> | Powerball | XML | Winning Numbers Beginning 2010 | * Date * Winning Numbers * Multiplier |
| [https://www.lotterycorner.com/ny/powerball/2019](https://www.lotterycorner.com/ny/powerball/2019" \t "_blank) | Powerball | HTML | NY Powerball 2019 Winning Numbers | * Date * Jackpot |

Examples of extracted data :

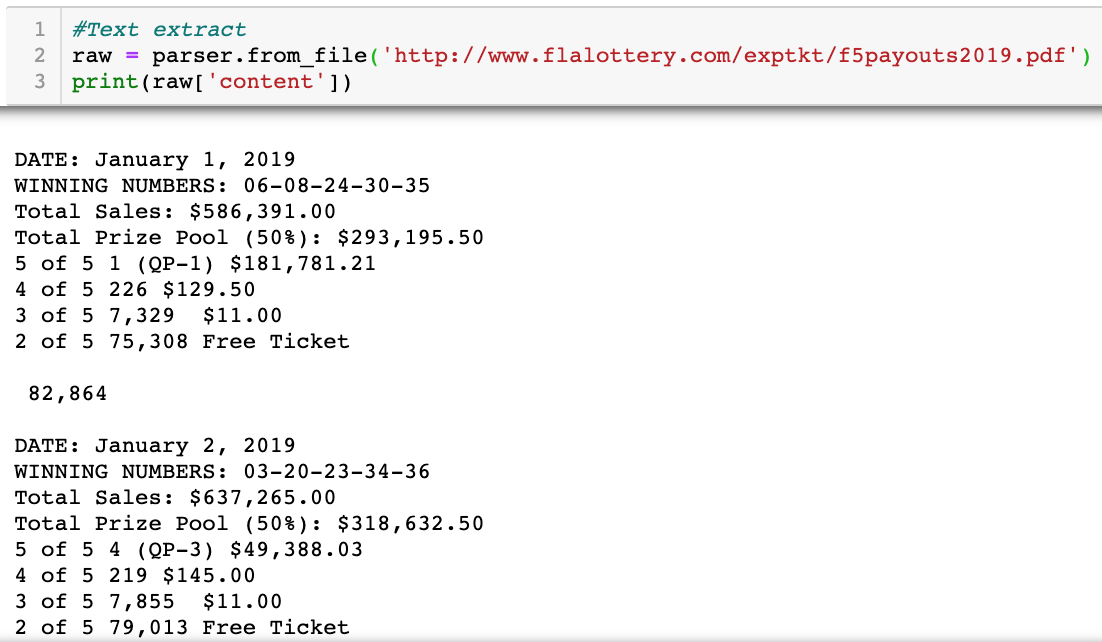
***Game : Florida Lotto***



***Game : Jackpot Triple Play***



Game : Fantasy 5



Game : Powerball



## Transform

As the pdf extracts did not contain tags or easy identifiable ways to get to the required fields, we implemented the python regular expressions to form patterns with which the required fields could be identified and written to lists.

Apart from this, the following transformations were performed on the different data sets:

|  |  |  |
| --- | --- | --- |
| Dataset | Field(s) | Transformation |
| Florida Lotto | Date | Formatted the date in the mm/dd/yyyy format |
| Date  Winning Numbers  Multiplier  Total Sales  Total Winners | Combined each list into one dataframe |
| Winning Numbers | Split the field into 5 columns each with a ball number  Renamed the columns to represent the balls |
| Total Sales | Removed the ‘$’ sign and changed to a float |
| Jackpot Triple Play | Date | Formatted the date in the mm/dd/yyyy format |
| Date  Winning Numbers  Total Sales  Prize Pool | Combined each list into one dataframe |
| Winning Numbers | Split the field into 6 columns each with a ball number  Renamed the columns to represent the balls |
| Total Sales | Removed the ‘$’ sign and ‘,’ and changed to a float |
| Prize Pool | Removed the ‘$’ sign and ‘,’ and changed to a float |
| Fantasy 5 | Date | Formatted the date in the mm/dd/yyyy format |
| Date  Winning Numbers  Total Sales | Combined each list into one dataframe |
| Winning Numbers | Split the field into 5 columns each with a ball number  Renamed the columns to represent the balls |
| Total Sales | Removed the ‘$’ sign and ‘,’ and changed to a float |
| Powerball (XML) | Multiplier | Added the multiplier tag in the rows that did not contain the multiplier and set the value to 0 |
|  | Date  Winning Numbers  Multiplier | Added the lists to a combined dataframe |
|  | Date | Formatted the date in mm/dd/yyyy format |
|  | Winning Numbers | Split the field into 5 columns each with a ball number  Added the Powerball in it’s own column  Renamed the columns to represent the balls |
| Powerball (PDF) | Date  Prize Payout (Actual Jackpot) | Removed data for unclaimed and expired prizes to ensure lists are the same count |
|  | Date  Advertized Jackpot  Prize Payout | Added the lists to a combined dataframe |
|  | Advertized Jackpot  Prize Payout | Removed the alpha numeric part that were used in the pattern to find the correct field  Remove the ‘$’, and ‘,’ and change to float |

## Load

All dataframes that were created were loaded to the MongoDB database called *Florida Lottery*. If the database does not exist, it will be created by the code. Same for the tables. Each dataframe is loaded into it’s own table. The following tables are created :

* fantasy5
* jackpot\_triple\_play
* lotto
* powerball
* powerball\_combined
* powerball\_winnings

We utilized a function, that was developed by a forum user, with some minor adjustments to load the data into the database.

The relational database was selected because the data we created is of a structure nature. Now that the tables are available the data can be used to create new datasets, graphs, or other statistical results.