**ETL Process Report**

***Introduction:***

For this project we decided to expand on our previous project and go further into sports betting. Instead of focusing on just soccer, we expanded and wanted to get data sets for different sports betting activities. We created a database with tables including betting data from Soccer, Horse racing, UFC and NFL.

***Extraction:***

-Extracted data from different csv files (sources below) using pandas read\_csv and an xlsx file using read\_xlsx.

<http://www.aussportsbetting.com/historical_data/nfl.xlsx>

<https://www.kaggle.com/mdabbert/ultimate-ufc-dataset?select=ufc-master.csv>

<https://www.kaggle.com/mdabbert/ufc-fights-2010-2020-with-betting-odds>

https://data.world/data-society/horse-racing-tipster-bets

***Transformation:***

- created a new column “New\_ID” for both datasets

-Copied columns from ufc betting data that were relevant and renamed the column as well – Red\_Corner\_Fighter, Blue\_Corner\_Fighter, Red\_Fighter\_odds, Blue\_Fighter\_odds, Winner, Red\_profit\_on\_a\_100\_credit\_winnin\_bet, "Blue\_profit\_on\_a\_100\_credit\_winnin\_bet"

-combined the two datasets into one dataset

**combined\_ufc\_df = pd.merge(ufc1\_dropcols, ufc2\_dropcols, how='outer', on='Fight\_ID')**

-Game score columns were replaced with a ‘Home Win’ Boolean column for NFL database

-In the end ‘Date’, ‘Home Team’, ’Away Team’, ‘Home Win’, ‘Home Odds Open’, ’Away Odds Open’ for NFL database

**Opening\_Odds\_df = Odds\_df[["Date", "Home Team", "Away Team", "Home Win", "Home Odds Open", "Away Odds Open"]]**

-Dropped columns "ID","Tipster","TipsterActive" from horseracing database

**horses2\_df=df.drop(columns=["ID","Tipster","TipsterActive",])**

-dropped the rows with NaN values

**Opening\_Odds\_df.dropna**

***Loading***

-Created database connection to postgres using SQLAlchemy ‘create\_engine’

**connection\_string = "postgres:samer123@localhost:5432/project2\_db"**

**engine = create\_engine(f'postgresql://{connection\_string}')**

-Confirmed table names from ETLProject\_db database created in postgres (see separate sql documentation for scheme set up)

-Loaded final pandas dataframes using ‘to\_sql’ method and connection created by SQLAlchemy dependency

**horses2\_df.to\_sql(name='horseracing',con=engine, if\_exists='replace', index=True)**

**ufc\_transformed.to\_sql(name='ufc\_transformed', con=engine, if\_exists='append', index=True)**

**odd\_transform.to\_sql(name='MatchOdd',con=engine, if\_exists='append', index=True)**

**result\_transform.to\_sql(name='MatchResult', con=engine, if\_exists='append', index=True)**

**top\_transform.to\_sql(name='BookiesTop5', con=engine, if\_exists='append', index=True)**

**Opening\_Odds\_df.to\_sql(name='nfl\_2006\_to\_2020\_odds', con=engine, if\_exists='append', index=True)**

***Limitations/Next Steps:***

***-***During the project there wasn’t too much limitations. We were able to successfully transform our data to show the data we hoped to show.

-We were successful in converting all our databases into one database in SQL to be able to quickly navigate between them and query each database.