Project 3

Alice Pao

Project Summary

This project is meant to use SQL to analyze differnet tables from a baseball database. It highlights baseball players who were once BYUI students; finding top 5 players having the highest batting average with at least 10 at bats of each year and also 100 at bats for their entire careers. It also compares Yankees and Red Sox's total winning games from 1900-2020.

Technical Detials

Grand Question 1

Create a new dataframe about baseball players who attended BYU-Idaho.

	playerID	schoolID	salary	yearID	teamID
0	lindsma01	idbyuid	4e+06	2014	CHA
1	lindsma01	idbyuid	3.6e+06	2012	BAL
2	lindsma01	idbyuid	2.8e+06	2011	COL
3	lindsma01	idbyuid	2.3e+06	2013	CHA
4	lindsma01	idbyuid	1.625e+06	2010	HOU
5	stephga01	idbyuid	1.025e+06	2001	SLN
6	stephga01	idbyuid	900000	2002	SLN
7	stephga01	idbyuid	800000	2003	SLN
8	stephga01	idbyuid	550000	2000	SLN
9	lindsma01	idbyuid	410000	2009	FLO
10	lindsma01	idbyuid	395000	2008	FLO

	playerID	schoolID	salary	yearID	teamID
11	lindsma01	idbyuid	380000	2007	FLO
12	stephga01	idbyuid	215000	1999	SLN
13	stephga01	idbyuid	185000	1998	PHI
14	stephga01	idbyuid	150000	1997	PHI

Grand Question 2a

Provide playerID, yearID, and batting average for players with at least 1 at bat that year.

	playerID	yearID	H	AB	batting_average
0	snowch01	1874	1	1	1
1	baldwki01	1884	1	1	1
2	mccafsp01	1889	1	1	1
3	gumbebi01	1893	1	1	1
4	oconnfr01	1893	2	2	1

Grand Question 2b

Provide playerID, yearID, and batting average for players with at least 10 at bat that year.

	playerID	yearID	H	AB	batting_average
0	nymanny01	1974	9	14	0.642857
1	carsoma01	2013	7	11	0.636364
2	altizda01	1910	6	10	0.6
3	johnsde01	1975	6	10	0.6
4	silvech01	1948	8	14	0.571429

Grand Question 2c

Calculate the batting average for players over their entire careers. (only for at least 100 at bats)

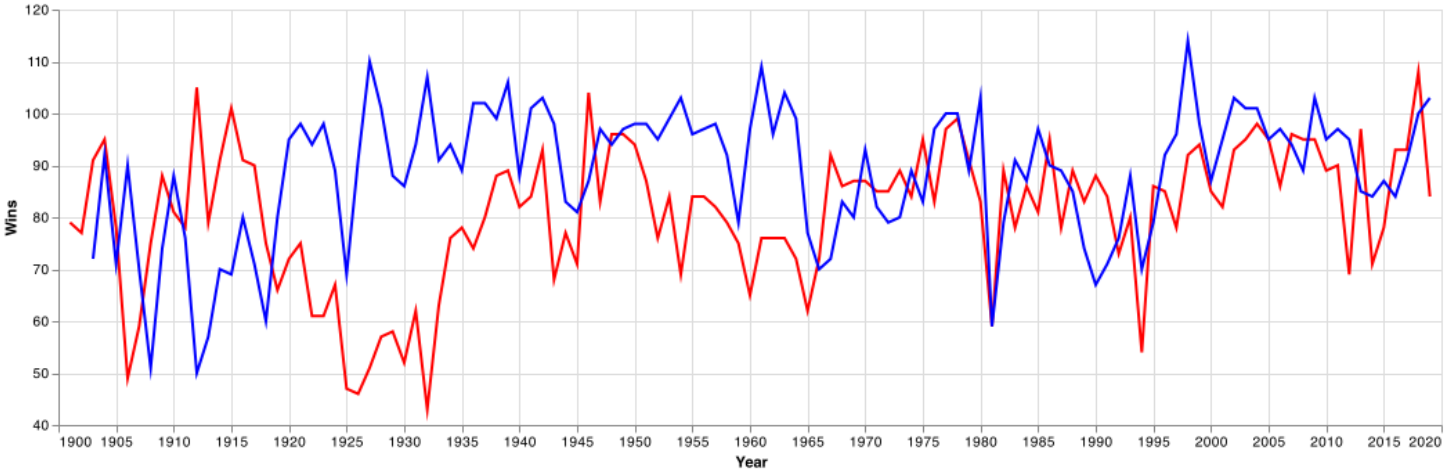
	playerID	yearID	H	AB	batting_average
0	hazlebo01	1957	54	134	0.402985
1	daviscu01	1939	40	105	0.380952
2	fishesh01	1930	95	254	0.374016
3	woltery01	1871	51	138	0.369565
4	cobbty01	1905	4189	11436	0.366299

Grand Question 3

Pick any two baseball teams and compare them using a metric of your choice. Make a graph to visualize the comparison

In this line chart, the red line represent Boston Red Sox and the blue line represent New York Yankees. The chart shows the number of wins these two teams have had from 1900-2020.

It shows that starting from 1918 to 1945, Yankees' has a lot more winning games than Red Sox.



Appendix A

```
#%%
import pandas as pd
import altair as alt
import sqlite3


baseball = sqlite3.connect('lahmansbaseballdb.sqlite')




# %%
# Grand Question 1
# Create a dataframe with the following attributes:
# playerID(People table), schoolID(Schools table), salary & yearID & teamID (Salaries table)

df1 = pd.read_sql_query("""SELECT DISTINCT cp.playerid, schoolid, salary, s.yearid, teamid
FROM salaries s
    JOIN collegeplaying cp
    ON s.playerid = cp.playerid
WHERE schoolid = 'idbyuid'
ORDER BY salary DESC
"""
,baseball)
print(df1.to_markdown())

# %%
# Grand Question 2
# a.

df2a = pd.read_sql_query("""
SELECT playerID, yearID, H, AB, (cast (H as float) / AB) as batting_average
FROM batting
ORDER BY batting_average DESC
LIMIT 5
""", baseball)
df2a.to_markdown()

# b.
df2b = pd.read_sql_query("""
SELECT playerID, yearID, H, AB, (cast (H as float) / AB) as batting_average
FROM batting
WHERE AB >= 10
ORDER BY batting_average DESC
LIMIT 5
""", baseball)
print(df2b.to_markdown())

# c.
df2c = pd.read_sql_query("""
SELECT playerID, yearID, SUM(H) as H, SUM(AB) as AB, (cast (SUM(H) as float) / SUM(AB)) as batting_aver
```

```
FROM batting
WHERE AB > 100
GROUP BY playerID
ORDER BY batting_average DESC
LIMIT 5
""", baseball)
print(df2c.to_markdown())

# %%
# Grand Question 3
df3_BOS = pd.read_sql_query("""
SELECT teamID, SUM(W) as W, SUM(L) as L, yearID
FROM teams
WHERE teamID = 'BOS'
GROUP BY yearID
""", baseball)
df3_BOS

df3_NYA = pd.read_sql_query("""
SELECT teamID, SUM(W) as W, SUM(L) as L, yearID
FROM teams
WHERE teamID = 'NYA'
GROUP BY yearID
""", baseball)
df3_NYA

df3_BOS_chart = (alt.Chart(df3_BOS)
    .properties(width = 1000)
    .mark_line(color = 'red')
    .encode(
        alt.Y('W', scale=alt.Scale(zero=False), title='Wins',),
        alt.X('yearID:Q', axis=alt.Axis(format='d'), title='Year', scale=alt.Scale(zero=False))
    )
)



df3_NYA_chart = (alt.Chart(df3_NYA)
    .properties(width = 1000)
    .mark_line(color = 'blue')
    .encode(
        y = alt.X('W', axis = alt.Axis(title = 'Wins'), scale = alt.Scale(zero = False)),
        x = alt.Y('yearID:Q', axis = alt.Axis(format = 'd', title = 'Year'), scale = alt.Scale(zero=Fal
    )
)

df3_combined_chart = df3_BOS_chart + df3_NYA_chart

df3_combined_chart.save('question_3.png')
```