Lab Exercise: Reading in database files

In [12]:

```
#Imports
import sqlite3 as sq3
import pandas as pd
```

- ### Create a variable, path, containing the path to the baseball.db contained in resources/

In [21]:

```
path = 'baseball[1].db'
path
```

Out[21]:

```
'baseball[1].db'
```

- ### Create a connection, con, that is connected to database at path

In [22]:

```
con = sq3.Connection(path)
con
```

Out[22]:

```
<sqlite3.Connection at 0x3b674e8>
```

- ### Create a variable, query, containing a SQL query which reads in all data from the allstarfull table

In [23]:

```
query = '''
SELECT *
    FROM allstarfull;
'''
query
```

Out[23]:

```
'\nSELECT *\n    FROM allstarfull;\n'
```

- ### Create a variable, observations, by using pandas' read_sql

In [25]:

```
observations = pds.read_sql(query,con)
observations.head()
```

Out[25]:

	index	playerID	yearID	gameNum	gameID	teamID	lgID	GP	startingPos
0	0	gomezle01	1933	0	ALS193307060	NYA	AL	1.0	1.0
1	1	ferreri01	1933	0	ALS193307060	BOS	AL	1.0	2.0
2	2	gehrilo01	1933	0	ALS193307060	NYA	AL	1.0	3.0
3	3	gehrich01	1933	0	ALS193307060	DET	AL	1.0	4.0
4	4	dykesji01	1933	0	ALS193307060	CHA	AL	1.0	5.0

index	playerID	yearID	gameNum	gameID	teamID	lgID	GP	startingPos

- ### Create a variable, tables, which reads in all data from the table sqlite_master

In [27]:

```
query = '''
SELECT *
FROM sqlite_master;
'''
all_tables = pd.read_sql(query, con)
all_tables
```

Out[27]:

	type	name	tbl_name	rootpage	sql
0	table	allstarfull	allstarfull	2	CREATE TABLE "allstarfull" (\n"index" INTEGER,...
1	index	ix_allstarfull_index	allstarfull	3	CREATE INDEX "ix_allstarfull_index"ON "allstar...
2	table	schools	schools	26	CREATE TABLE "schools" (\n"index" INTEGER,\n ...
3	index	ix_schools_index	schools	31	CREATE INDEX "ix_schools_index"ON "schools" ("...
4	table	batting	batting	99	CREATE TABLE "batting" (\n"index" INTEGER,\n ...
5	index	ix_batting_index	batting	100	CREATE INDEX "ix_batting_index"ON "batting" ("...

- ### Pretend that you were interesting in creating a new baseball hall of fame. Join and analyze the tables to evaluate the top 3 all time best baseball players.

In [30]:

```
best_query = """
SELECT playerID, sum(GP) AS num_games_played, AVG(startingPos) AS avg_starting_position
    FROM allstarfull
    GROUP BY playerID
    ORDER BY num_games_played DESC, avg_starting_position ASC
    LIMIT 3
"""
best = pd.read_sql(best_query, con)
best.head()
```

Out[30]:

	playerID	num_games_played	avg_starting_position
0	musiast01	24.0	6.357143
1	mayswi01	24.0	8.000000
2	aaronha01	24.0	8.470588

In [ ]:

```

```