

baseball_mysql

February 10, 2018

```
In [1]: import pymysql.cursors
```

```
In [2]: conn = pymysql.connect(host='localhost',
                                user='root',
                                password='',
                                db='baseball',
                                charset='utf8',
                                cursorclass=pymysql.cursors.DictCursor)

    cursor = conn.cursor()
```

```
In [3]: sql = """CREATE TABLE batting(
            id int(11) NOT NULL AUTO_INCREMENT,
            player_id VARCHAR(20) NOT NULL,
            year INT NOT NULL,
            team_id VARCHAR(20) NOT NULL,
            games INT NOT NULL,
            at_bats INT NOT NULL,
            runs_scored INT NOT NULL,
            hits INT NOT NULL,
            homeruns INT NOT NULL,
            runs_batted_in INT NOT NULL,
            walks INT NOT NULL,
            strike_outs INT NOT NULL,
            PRIMARY KEY (id)) AUTO_INCREMENT=1;"""
    cursor.execute(sql)
```

```
Out[3]: 0
```

```
In [4]: sql = """CREATE TABLE fame(
            id int(11) NOT NULL AUTO_INCREMENT,
            player_id VARCHAR(20) NOT NULL,
            year INT NOT NULL,
            voted_by VARCHAR(20) NOT NULL,
            ballots INT NOT NULL,
            needed INT NOT NULL,
            votes INT NOT NULL,
            inducted CHAR(1) NOT NULL,
            category VARCHAR(20) NOT NULL,
```

```
PRIMARY KEY (id)) AUTO_INCREMENT=1;"""
cursor.execute(sql)
```

Out[4]: 0

```
In [5]: sql = """CREATE TABLE player(
        player_id VARCHAR(20) NOT NULL,
        first_name VARCHAR(50) NOT NULL,
        last_name VARCHAR(50) NOT NULL,
        PRIMARY KEY (player_id));"""
cursor.execute(sql)
```

Out[5]: 0

```
In [6]: import csv
```

```
In [7]: with open('Batting.csv') as file:
        reader = csv.DictReader(file)
        for i, row in enumerate(reader):
            id = row['playerID']
            year = int(row['yearID'])
            team_id = row['teamID'].replace("'", '')
            games = int(row['Games'])
            at_bats = int(row['Atbats'])
            runs_scored = int(row['RunsScored'])
            hits = int(row['Hits'])
            homeruns = int(row['Homeruns'])
            runs_batted_in = int(row['RunsBattedIn'])
            walks = int(row['Walks'])
            strike_outs = int(row['strikeOuts'])
            sql = f"""INSERT INTO batting
            (player_id,year,team_id,games,at_bats,runs_scored,hits,homeruns,runs_batted_in,w
            VALUES
            ({id},{year},{team_id},{games},{at_bats},{runs_scored},{hits},{homeruns},
            {runs_batted_in},{walks},{strike_outs});"""
            cursor.execute(sql)
```

```
In [8]: with open('HallofFame.csv') as file:
        reader = csv.DictReader(file)
        for i, row in enumerate(reader):
            id = row['playerID']
            year = int(row['yearid'])
            voted_by = row['votedBy'].replace("'", '')
            ballots = int(row['ballots'])
            needed = int(row['needed'])
            votes = int(row['votes'])
            inducted = row['inducted'].replace("'", '')
            category = row['category'].replace("'", '')
            sql = f"""INSERT INTO fame
```

```

        (player_id,year,voted_by,ballots,needed,votes,inducted,category)
VALUES
('{id}',{year},{voted_by},{ballots},{needed},{votes},{inducted},{category}')
cursor.execute(sql)

```

```

In [9]: with open('Players.csv') as file:
        reader = csv.DictReader(file)
        for i, row in enumerate(reader):
            id = row['playerid']
            first = row['namefirst'].replace("'", "'")
            last = row['namelast'].replace("'", "'")
            sql = f"""INSERT INTO player
            (player_id,first_name,last_name)
VALUES
('{id}','{first}','{last}');"""
            cursor.execute(sql)

```

```

In [10]: conn.commit()

```

1a) how many players are inducted into the hall of fame?

```

In [11]: sql = "SELECT COUNT(DISTINCT player_id) _no FROM fame WHERE inducted='Y' AND category = 'Player'"
        cursor.execute(sql)
        one = cursor.fetchone()
        one.get('_no')

```

```

Out[11]: 247

```

1b) how many distinct players in the batting data set are in the hall of fame?

```

In [29]: sql = """SELECT COUNT(DISTINCT b.player_id) _no
        FROM batting b
        WHERE b.player_id IN (SELECT f.player_id FROM fame f
        WHERE f.category = 'Player' AND f.inducted = 'Y')"""
        cursor.execute(sql)
        one = cursor.fetchone()
        one.get('_no')

```

```

Out[29]: 221

```

2a) what player has exactly 3000 hits total for his career?

```

In [14]: sql = """SELECT player_id FROM batting GROUP BY player_id HAVING SUM(hits) = 3000"""
        cursor.execute(sql)
        one = cursor.fetchone()
        one.get('player_id')

```

```

Out[14]: 'clemero01'

```

2b) what year was he inducted into the hall of fame?

```
In [15]: sql = """SELECT year FROM fame WHERE inducted = 'Y' AND player_id =
          (SELECT player_id FROM batting GROUP BY player_id HAVING SUM(hits) = 3000)
          ORDER BY year"""
          cursor.execute(sql)
          one = cursor.fetchone()
          one.get('year')
```

Out[15]: 1973

3) what player had the highest number of homeruns in 1919?

```
In [16]: sql = """SELECT player_id FROM batting WHERE year=1919 GROUP BY player_id ORDER BY sum(hits)
          cursor.execute(sql)
          one = cursor.fetchone()
          one.get('player_id')
```

Out[16]: 'ruthba01'

4) what team has had the most hits after the year 2000, how many?

```
In [17]: sql = """SELECT team_id,SUM(hits) _no FROM batting WHERE year > 2000 GROUP BY team_id ORDER BY SUM(hits)
          cursor.execute(sql)
          one = cursor.fetchone()
          one
```

Out[17]: {'_no': Decimal('23029'), 'team_id': 'BOS'}

5a) what player has the most cumulative hits?

```
In [18]: sql = """SELECT player_id,SUM(hits) _no FROM batting GROUP BY player_id ORDER BY SUM(hits)
          cursor.execute(sql)
          one = cursor.fetchone()
          player_id = one.get('player_id')
          player_id
```

Out[18]: 'rosepe01'

5b) how many does he have?

```
In [19]: one.get('_no')
```

Out[19]: Decimal('4256')

5c) is he in the hall of fame?

```
In [20]: sql = f"""SELECT COUNT(*) _no FROM fame WHERE player_id = '{player_id}' AND inducted = 'Y'
          cursor.execute(sql)
          one = cursor.fetchone()
          if one.get('_no') > 0:
              print("Yes")
          else:
              print("No")
```

No

5d) how many votes has he received total?

```
In [21]: sql = f"""SELECT SUM(votes) _no FROM fame WHERE player_id = '{player_id}'"""
         cursor.execute(sql)
         one = cursor.fetchone()
         one.get('_no')
```

```
Out[21]: Decimal('74')
```

6a) what 3 players have the highest number of cumulative strikeouts over their career that have been inducted into the hall of fame?

6b) how many total strikeouts do they each have?

```
In [23]: sql = """SELECT b.player_id,SUM(b.strike_outs) _outs
FROM batting b
WHERE b.player_id IN
(SELECT DISTINCT f.player_id FROM fame f WHERE f.inducted = 'Y')
GROUP BY b.player_id
ORDER BY SUM(b.strike_outs) DESC LIMIT 3"""
         cursor.execute(sql)
         for row in cursor.fetchall():
             print(row.get('player_id'), row.get('_outs'))
```

```
jacksre01 2597
stargwi01 1936
schmimi01 1883
```

7a) what 3 players have the highest cumulative total number of homeruns and have been inducted into the hall of fame?

```
In [24]: sql = """SELECT b.player_id,SUM(b.homeruns) _homeruns
FROM batting b
WHERE b.player_id IN
(SELECT f.player_id FROM fame f WHERE f.inducted = 'Y')
GROUP BY b.player_id
ORDER BY SUM(b.homeruns) DESC LIMIT 3"""
         cursor.execute(sql)
         for row in cursor.fetchall():
             print(row.get('player_id'), row.get('_homeruns'))
```

```
aaronha01 755
ruthba01 714
mayswi01 660
```

7b) what 3 players have the highest cumulative total number of homeruns and have NOT been inducted into the hall of fame?

```
In [25]: sql = """SELECT b.player_id,SUM(b.homeruns) _homeruns
FROM batting b
WHERE b.player_id NOT IN
(SELECT f.player_id FROM fame f WHERE f.inducted = 'Y')
GROUP BY b.player_id
ORDER BY SUM(b.homeruns) DESC LIMIT 3"""
cursor.execute(sql)
for row in cursor.fetchall():
    print(row.get('player_id'), row.get('_homeruns'))
```

bondsba01 762

rodrial01 687

thomeji01 612

8) what player had the highest batting average from 1930 to 1940 with a cumulative total of hits over 1800 and has NOT been elected into the hall of fame? batting average = hits/atbats

```
In [26]: sql = """SELECT player_id,SUM(hits),SUM(hits)/SUM(at_bats) _avg
FROM batting
WHERE
player_id NOT IN (SELECT player_id FROM fame WHERE inducted = 'Y')
AND (year >= 1930 and year <= 1940)
GROUP BY player_id
HAVING SUM(hits) > 1800
ORDER BY _avg DESC LIMIT 1"""
cursor.execute(sql)
one = cursor.fetchone()
one.get('player_id')
```

Out[26]: 'chapmbe01'

9) what player averaged the highest number of walks from 1900-1920 with over 500 atbats?

```
In [27]: sql = """SELECT player_id
FROM batting
WHERE (year >= 1900 and year <= 1920)
GROUP BY player_id
HAVING sum(at_bats) > 500
ORDER BY SUM(walks)/COUNT(DISTINCT year) DESC
LIMIT 1"""
cursor.execute(sql)
one = cursor.fetchone()
one.get('player_id')
```

Out[27]: 'hamilbi01'

```
In [28]: sql = """SELECT player_id
FROM
(SELECT player_id,year,sum(at_bats) _at_bats,MAX(walks) _walks
FROM batting
GROUP BY player_id,year) t
WHERE (year >= 1900 and year <= 1920)
GROUP BY player_id
HAVING sum(_at_bats) > 500
ORDER BY SUM(_walks)/COUNT(DISTINCT year) DESC
LIMIT 1"""
cursor.execute(sql)
one = cursor.fetchone()
one.get('player_id')

Out[28]: 'hamilbi01'
```