

1. webシステム基礎実験（JavaScriptクラス）本格的なDBの使い方編

今回は、テーブルを連結して、多様なデータを扱うことを目標とする。

1.1. 準備

いつもどおりPaiza.cloudを使用する。サーバを起動する際、Node.jsとMySQLを使用するようにチェックを入れること。

起動したらターミナルから以下のように順番に入力して、必要なデータをセットしよう。

まずはリポジトリのcloneを行う。

```
~$ git clone https://github.com/sudahiroshi/websystem3.git
Cloning into 'websystem3'...
remote: Enumerating objects: 12, done.
create table player ( id int auto_increment not null primary key, name
varchar(1remote: Counting objects: 100% (12/12), done.
remote: Compressing objects: 100% (9/9), done.
remote: Total 12 (delta 1), reused 9 (delta 1), pack-reused 0
Unpacking objects: 100% (12/12), done.
~$
```

ディレクトリを変更する。

```
~$ cd websystem3
~/websystem3$
```

DBの初期化を行う。

```
~/websystem3$ sudo mysql < init.sql
~/websystem3$
```

データの流し込みを行う。このとき、Warningが発生しているが、今は気にしなくて良い。

```
~/websystem3$ mysql -u node -pwebsystem web < db.sql
mysql: [Warning] Using a password on the command line interface can be
insecure.
~/websystem3$
```

1.2. DBの確認

ここまできちんとできていることを、データを閲覧して確認してみよう。今回例として使用したのは日本プロ野球の打撃成績である。年度ごとの打撃成績、選手名、球団名をそれぞれテーブルとしている。

まずはmysqlコマンドの起動。

```
~/websystem3$ mysql -u node -pwebsystem web
mysql: [Warning] Using a password on the command line interface can be insecure.
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 4
Server version: 5.7.24-0ubuntu0.18.04.1 (Ubuntu)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

続いて、選手の一覧。昔の選手が多いのは気にしないでください。

```
mysql> select * from player;
+----+-----+
| id | name      |
+----+-----+
|  1 | イチロー  |
|  2 | 秋山幸二  |
|  3 | 落合博満  |
+----+-----+
3 rows in set (0.00 sec)

mysql>
```

続いて、球団一覧。上記の選手が所属したことのある球団のみ登録してあります。

```
mysql> select * from team;
+----+-----+
| id | name      |
+----+-----+
|  1 | オリックス |
+----+-----+
```

```

| 2 | 西武 |
| 3 | 福岡ダイエー |
| 4 | ロッテ |
| 5 | 中日 |
| 6 | 読売 |
| 7 | 日本ハム |
+-----+
7 rows in set (0.00 sec)

mysql>

```

続いて打撃成績。ここでは最初の10個分のデータのみ表示しています。

```

mysql> select * from batting limit 10;
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | year | player_id | team_id | PA | AB | H | HR | R |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | 1992 | 1 | 1 | 99 | 95 | 24 | 0 | 5 |
| 2 | 1993 | 1 | 1 | 67 | 64 | 12 | 1 | 3 |
| 3 | 1994 | 1 | 1 | 616 | 546 | 210 | 13 | 54 |
| 4 | 1995 | 1 | 1 | 613 | 524 | 179 | 25 | 80 |
| 5 | 1996 | 1 | 1 | 611 | 542 | 193 | 16 | 84 |
| 6 | 1997 | 1 | 1 | 607 | 536 | 185 | 17 | 91 |
| 7 | 1998 | 1 | 1 | 558 | 506 | 181 | 13 | 71 |
| 8 | 1999 | 1 | 1 | 468 | 411 | 141 | 21 | 68 |
| 9 | 2000 | 1 | 1 | 459 | 395 | 153 | 12 | 73 |
| 10 | 1981 | 2 | 2 | 6 | 5 | 1 | 0 | 0 |
+----+-----+-----+-----+-----+-----+-----+-----+
10 rows in set (0.00 sec)

mysql>

```

battingの項目は以下の通り。

項目名	意味
id	ID
yser	年度
player_id	上の方に書いた選手テーブルのid
team_id	上の方に書いた球団テーブルのid
PA	打席数
AB	打数
H	安打数
HR	ホームラン数

項目名	意味
R	打点

1.3. 表の結合

このままでは、とても見づらい出力結果になっているので、複数のテーブルを結合して見やすくしよう。

まずは、選手名をきちんと表示させる例を示す。ここで、`innerjoin`が表の結合を行うための単語で、`inner join`をつなげた単語である。SQLにおいて、表の結合はいくつかの種類があり、最もよく使うのが `inner join` である。使い方であるが、`from batting innerjoin player`のように`from`の後ろにメインとなるテーブル名を書き、その後に`innerjoin`と結合するテーブルを記述する。

そして、最も重要なのがその後にある`on`の項目である。この例では`on batting.player_id = player.id`となっている。これは、`batting`テーブルの`player_id`と`player`テーブルの`id`が同じになるように結合することを意味している。

また、`select`の後の、表示項目欄にテーブル名が付いている。これは、複数のテーブルからデータを取得するので、テーブル名がないとどの項目かわからなくなってしまうのを防ぐためである。

```
mysql> select batting.id, batting.year, batting.HR, player.name
       from batting
       innerjoin player on batting.player_id = player.id
       limit 10;
```

id	year	HR	name
1	1992	0	イチロー
2	1993	1	イチロー
3	1994	13	イチロー
4	1995	25	イチロー
5	1996	16	イチロー
6	1997	17	イチロー
7	1998	13	イチロー
8	1999	21	イチロー
9	2000	12	イチロー
10	1981	0	秋山幸二

10 rows in set (0.00 sec)

```
mysql>
```

1.4. 練習問題

上の例のように、打撃成績のid、年度、ホームラン数と球団名を表示するqueryを記述せよ。なお、選手名は不要である。

1.5. 3つの表の結合

続いて、3つのテーブルを結合してみよう。その場合、`innerjoin`と`on`の組が増えることとなる。具体的な例を以下に示す。

```
mysql> select batting.id, batting.year, batting.HR, player.name, team.name
       from batting
       inner join player on batting.player_id = player.id
       inner join team on batting.team_id = team.id
       limit 10;
```

id	year	HR	name	name
1	1992	0	イチロー	オリックス
2	1993	1	イチロー	オリックス
3	1994	13	イチロー	オリックス
4	1995	25	イチロー	オリックス
5	1996	16	イチロー	オリックス
6	1997	17	イチロー	オリックス
7	1998	13	イチロー	オリックス
8	1999	21	イチロー	オリックス
9	2000	12	イチロー	オリックス
10	1981	0	秋山幸二	西武

10 rows in set (0.00 sec)

mysql>

1.6. 集計

SQLには、最大値を取得する関数が備わっている。使用例を以下に示す。ここでは、ホームラン数（項目名：HR）の最大値を調べている。

```
mysql> select max(HR) from batting;
```

max(HR)
52

1 row in set (0.00 sec)

mysql>

さて、年間52本のホームラン数を記録したのは誰でしょう？これを調べるためには、まだまだSQLの詳しい使い方を知る必要があるので、後回しにします。

続いて、playerごとの最大ホームラン数を調べてみましょう。ここに出てくる単語が`group by`です。集計時に`group by`の後ろにまとめたい項目名を書いておきます。ここでは、`player_id`ごとに最大値を求めるというqueryになります。

```
mysql> select player_id,max(HR) from batting group by player_id;
+-----+-----+
| player_id | max(HR) |
+-----+-----+
|          1 |        25 |
|          2 |        43 |
|          3 |        52 |
+-----+-----+
3 rows in set (0.00 sec)

mysql>
```

1.7. 練習問題

球団ごとの最大ホームラン数を調べるqueryを記述せよ。なお、選手名も球団名も不要で、IDが表示されれば良い。