

セッション 7-ハンズオンスクリプト

■【回答①】人事情報データベース作成

このレクチャーでは、人事情報データベースを作成していただきます。

データベース名とテーブル名ですが、実際にシステム上に登録する上では、日本語名ではなくアルファベットで記載する方が良いとされています。アルファベットの方がシステム言語に近い形態であるため、それによって認識が容易であるためです。ですので今回は人事情報データベースを「hr_data」というデータベース名で作成していきます。

同じように以降のテーブル作成においてもテーブル名はアルファベットで作成させていただきます。

それではコマンドを入力していきましょう。

```
CREATE DATABASE hr_data;
```

のコマンドでデータベースを作ってみます。

それでは実行してみます。

正常に実行できたことを確認して、

```
show databases
```

コマンドで見te確認していただくと、先ほど追加した

HRDB が追加されていることがわかります。

■【回答2】社員情報テーブルの作成

このレクチャーでは先ほど作った ht_data データベース内に社員情報テーブルを作成します。

次にそのテーブルに実際に社員情報を追加していきます。

まずは

```
use hr_data;
```

を実行していただき、先ほどのデータベースを指定します。

次に

```
show tables;
```

を実行していただき、現在のテーブル一覧を確認してみましょう。

まだ何もテーブルがありません。

それでは、社員情報を管理する社員情報テーブルを作成してみましょう。

ここでは作成するテーブル名を

```
employees
```

とします。

データ項目はそれぞれ

- 社員 ID は employee_id
- 姓は last_name
- 名は first_name
- 性別は sex
- 電話番号 phone
- 入社年月日は joining_date
- 年齢は birthday
- 所属部署は department

としていきます。

データベースの項目は日本語ではなく、アルファベットで設定することが基本となりますので、英語名やローマ字などを利用して明確にデータ項目がわかりやすい名前で設定するようにしてください。

それではテーブルを作成していきます。

```
CREATE TABLE employees  
(
```

```
employee_id INT(6) PRIMARY KEY,  
last_name VARCHAR (15) NOT NULL,  
first_name VARCHAR(15) NOT NULL,  
sex VARCHAR(1) NOT NULL,  
phone VARCHAR(13) NOT NULL UNIQUE KEY,  
joining_date DATE NOT NULL,  
bithday date NOT NULL,  
department VARCHAR(30) NOT NULL DEFAULT "unassigned"  
)ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

一番最後に ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;を追加しています。今回は初めて日本語データを利用しますので、こちらを設定いします。

そして、今回は名字をインデックス設定すると問題にあったので、そちらの設定を行なっていきます。

下記のコマンドを記入して実行します。

```
CREATE INDEX index_name ON employees(last_name);
```

これで、インデックス設定できました。

```
show tables;
```

で現在のテーブル一覧を確認してみましょう。

先ほど作ったテーブルが作成されていることを確認できます

それでは 3 人分のデータを追加していきたいと思います。今回追加する 3 人のデータをご説明します。こちらの内容は添付テキストファイルに記載されていますので、そちらをご確認ください。

それでは入力していきます。

```
INSERT INTO employees
(employee_id, last_name, first_name, sex, phone, joining_date, age, department)
VALUES
(1001,'佐藤','隆','m','060-9783-0193',20180401,23,'営業'),
(1002,'遠藤','真紀','w','060-8503-1284',20160401,24,'人事'),
(1003,'工藤','孝明','m','060-9874-6382',20100401,30,'開発');
```

とデータを見やすく設定していきましょう。

実行してみます。

正常に実行できました。

```
SELECT * FROM employees;
```

で中身が正常に登録されているかを確認していきましょう。

これで完了となります。

■【回答3】人事評価テーブルの作成

このレクチャーでは hr_data データベース内に人事評価テーブルを作成します。

次にそのテーブルに実際に給与情報を追加していきます。

ここでは作成するテーブル名を evaluations とします。

データ項目はそれぞれ

- 整理番号 は id として 1 から自動採番（プライマリーキーとして設定）
- 社員 ID は employee_id を外部キーで設定する
- 評価年度は year
- 評価ランクは rating A~E で評価します。
- 評価内容は explanation

すべての項目に Not Null 制約を設定します。

また、親テーブルの employee_id が削除/更新された際に、子テーブルにも反映させるために DELETE CASCADE と UPDATE CASCADE の設定も行なっておきます。

最後に、chara 型を日本語に対応しておくオプションも設定しておきます。

それではテーブルを作成していきます。

```
CREATE TABLE evaluations
(
    id INT(6) PRIMARY KEY AUTO_INCREMENT,
    employee_id INT(6) NOT NULL,
    year YEAR NOT NULL,
    rating VARCHAR(1) NOT NULL,
    explanation VARCHAR(1000) NOT NULL,
    FOREIGN KEY (employee_id) REFERENCES employees (employee_id)
)ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

SHOW tables;

で現在のテーブル一覧を確認してみましょう。

念のためにオプション設定が正しくなされたかを確認するためにインフォメーションを見ておきましょう。Evaluations テーブルのインフォメーションのカラムを確認します。

すべての項目が Not null の設定がなされていることを確認してください。

また、外部キーも正しく設定されています。

先ほど作ったテーブルが作成されていることを確認できます。

それでは 3 人分のデータを追加していきたいと思います。

1 人目は

- 整理番号は自動採番
- 社員 ID 1001
- 評価年度は 2019 年
- 評価ランクは A
- 評価内容は“非常に優秀でした。”

2 人目は

- 整理番号は自動採番
- 社員 ID 1002
- 評価年度は 2019 年
- 評価ランクは B
- 評価内容は“良かったです。”

3 人目は

- 整理番号は自動採番
- 社員 ID 1003
- 評価年度は 2019 年

- 評価ランクは E
- 評価内容は“今後の改善が必要です。”

それでは入力していきます。

```
INSERT INTO evaluations
(employee_id, year, rating, explanation)
VALUES

(1001,2019, 'A','非常に優秀でした。'),

(1002,2019, 'B','良かったです。'),

(1003,2019, 'E','今後の改善が必要です。');
```

整理番号は自動採番されるように、入力しないでおきたいと思います。もちろん、こちらで設定をしてもらっても構いません。

実行してみます。

正常に実行できました。

```
SELECT * FROM evaluations;
```

で中身が正常に登録されているかを確認していきましょう。

これで完了となります。

w

■【回答 4】給与情報テーブルの作成

このレクチャーでは hr_data データベース内に給与情報テーブルを作成します。

次にそのテーブルに実際に給与情報を追加していきます。

それでは、社員情報を管理する社員情報テーブルを作成してみましょう。

ここでは作成するテーブル名を payrolls とします。

データ項目はそれぞれ

- 整理番号 は id として 1 から自動採番（プライマリーキーとして設定）
- 社員 ID は employee_id を外部キーで設定する
- 役職は position
- 標準月額給与決定年月日は decided_date
- 標準月額給与は payroll_rate

としていきます。

それではテーブルを作成していきます。

```
CREATE TABLE payrolls  
(
```

```
id INT(6) PRIMARY KEY AUTO_INCREMENT,  
employee_id INT(6) NOT NULL,  
position VARCHAR(20) NOT NULL,  
decided_date DATE NOT NULL,  
payroll_rate INT(20) NOT NULL,  
FOREIGN KEY (employee_id) REFERENCES employees (employee_id)  
)ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

入力ができたら実行してみます。

```
SHOW tables;
```

で現在のテーブル一覧を確認してみましょう。

先ほど作ったテーブルが作成されていることを確認できます

それでは 3 人分のデータを追加していきたいと思います。

それでは入力していきます。

```
INSERT INTO payrolls  
(employee_id, position, decided_date, payroll_rate)  
VALUES  
  
(1001,'営業担当',20190301,200000),  
  
(1002,'リーダー',20190301,250000),  
  
(1003,'係長',20190301,300000);
```

実行してみます。

正常に実行できました。

```
SELECT * FROM payrolls;
```

で中身が正常に登録されているかを確認していきましょう。