

Webプログラミング

DataBase 2

- MySQL編 -

山崎 大助

D.Yamazaki

アジェンダ

- 新しく学ぶこと

Databaseを操作する

- SELECT

- UPDATE

- DELETE

- 課題

データベース作成

Database作成（XAMPP/MAMP使用した場合）

1. 最初にデータベースを作成します。

MAMP: <http://localhost/phpMyAdmin/index.php> (大文字に注意！)

XAMPP: <http://localhost/phpmyadmin/index.php>

2. 画面に左のメニュー “phpMyAdmin” を選択

3. 『データベース』 タブをクリック

4. 『データベースを作成する』 に半角英数で任意のデータベース名、
「gs_db」を入力し作成。

5. 『照合順序』は utf8_unicode_ci を選択。

6. 『作成』 ボタンで作成。



アンケートシステムのDB構築

◇テーブル作成

DB名: gs_db

Table名: gs_an_table

Field名:

id:	int(12)	<u>AUTO INCREMENT PRIMARY KEY</u>
name :	var_char(64)	
email :	var_char(256)	
naiyou :	text	
age:	int(3)	
indate:	datetime	

※フォームの項目を増やした場合こちらのFieldも増やしましょう。

※Fieldの右にあるのは、データ型(Type)です。

<http://mysql.akarukutanoshiku.com/category5/entry21.html>

データ登録

テーブル基本操作：データベース言語 SQL

MySQLは以下のコマンドを使用してデータ登録・更新・削除が可能です。

◇ SQLコマンド（データ操作）

INSERT： データを“登録”する事ができます。

SELECT： データを“表示”する事ができます。

UPDATE： データを“更新”する事ができます。

DELETE： データを“削除”する事ができます。

◇ SQLのコメント

例： -- SELECT * FROM テーブル名;

“--”ハイフンを連続2文字並べることで次の文字からコメントになります。

◇ SQLで文字列を扱う

シングルクォートで囲む。 ダブルクォートはNG。

テーブル基本操作 SQL

■ INSERT (データ登録)

◇書式：

```
INSERT INTO テーブル名(カラム1, カラム2,...)
VALUES(値1, 値2,...);
```

```
INSERT INTO gs_an_table(id, name, email, age, naiyou, indate)
VALUES(NULL, 'ジーズ太郎', 'test1@test.test', 20, 'テスト1', '2020-06-15
00:00:00');
```

```
INSERT INTO gs_an_table(id, name, email, age, naiyou, indate)
VALUES(NULL, 'ジーズ次郎', 'test2@test.test', 10, 'テスト2', sysdate() );
```

※文字列を登録する場合シングルクォートで囲んで指定します。

データ取得

テーブル基本操作 SQL

■ SELECT (データ取得)

◆書式：

SELECT 表示するカラム FROM テーブル名;

```
SELECT * FROM gs_an_table; --全指定
```

```
SELECT name FROM gs_an_table; --単体指定
```

```
SELECT name, email FROM gs_an_table;    --複数指定
```

```
SELECT * FROM gs_an_table WHERE name = 'ジーズ太郎';
```

※WHERE を使用して特定のデータを抽出することが可能です。

条件付き検索

◇以下条件付きSQLの一例)

- ・ 演算子を使う

SELECT * FROM テーブル名 WHERE id = 1;

SELECT * FROM テーブル名 WHERE id >= 3;

- ・ AND, OR で検索条件を複数指定する

SELECT * FROM テーブル名 WHERE id = 1 OR id = 2;

SELECT * FROM テーブル名 WHERE id >= 1 AND id <= 3;

- ・ あいまい検索をする

SELECT * FROM テーブル名 WHERE indate LIKE '2020-06%';

SELECT * FROM テーブル名 WHERE email LIKE '%@gmail.com';

SELECT * FROM テーブル名 WHERE email LIKE '%@%';

ソートと制限

◇表示をソートする

書式：

SELECT * FROM テーブル名 ORDER BY ソートしたいカラム名 ***;

SELECT * FROM テーブル名 ORDER BY id DESC;

SELECT * FROM テーブル名 ORDER BY email, name DESC;

※ DESC は降順でSORT、ASC は昇順でSORT

◇表示件数を制限する

書式：

SELECT 表示するカラム FROM テーブル名 LIMIT ***;

SELECT * FROM テーブル名 LIMIT 5;

SELECT * FROM テーブル名 LIMIT 3, 5;

データ更新

テーブル基本操作 SQL

■ UPDATE (データ更新)

◇書式：

UPDATE テーブル名 SET 変更データ WHERE 選択データ;

```
UPDATE gs_an_table SET  
    name= 'ブート太郎' WHERE id= 1;
```

```
UPDATE gs_an_table SET  
    name= 'ブート次郎' WHERE id= 2;
```

※ WHERE を使用して特定のデータを更新します→忘れると全てのデータが更新されるので知っておきましょう。

データ削除

テーブル基本操作 SQL

■ DELETE (データ削除)

◇書式：

DELETE FROM テーブル名;

--全て削除

DELETE FROM gs_an_table;

--WHEREを使用して特定のデータを削除します。

DELETE FROM gs_an_table **WHERE id=1;**

注意) DELETEするとデータは復旧できないので気をつけてましょう。

課題

アンケートシステムのDB構築



作成済み

◇テーブル作成

DB名: gs_db

Table名: gs_an_table

Field名:

id:	int(12)	<u>AUTO INCREMENT PRIMARY KEY</u>
name :	var_char(64)	
email :	var_char(256)	
naiyou :	text	
age:	int(3)	
indate:	datetime	

★POINT:

テストデータ10個入れておきましょう！

“age”列には[10, 20, 30, 40,50]のどれかを入れておきましょう！

課題: SQLをつくってください

使用 DB名: **gs_db** Table名: **users**

配布資料「**SQL課題.txt**」以下SQL文を作成して提出

1. SELECT文を使って、"id" 「1,3,5」 だけ抽出するSQLを作る
2. SELECT文を使って、"id" 「4～8」 を抽出するSQLを作る
3. SELECT文を使って、"email" 「 test1」 を抽出するあいまい検索を作る
4. SELECT文を使って、"新しい日付順"にソートするSQLを作る。
5. SELECT文を使って、"age" 「20」 で"indate" 「2020-06%」 のデータを抽出するSQLを作る
6. SELECT文を使って、"新しい日付順"で、「5個」 だけ取得するSQLを作る
7. (難問題) "age"で「GROUP BY 」使い10,20,30,40歳が各何人知るか抽出するSQLを作る

	age	count(*)
1	10	2
2	20	4
3	30	3
	40	1

授業制作物：提出の方法

1. php05フォルダを新規作成する。
2. php05フォルダに「 **SQL課題.txt** 」を入れる。
3. デスクトップの”gs_code”フォルダにphp05フォルダを移動
4. Githubへプッシュ(アップロード)すればOK

締め切り:

翌週授業(授業開始時)まで