

DBを理解しよう

ID	Name	age
1	山田太郎	23

この教材でできること

- DBについて理解できる
- Pythonで出力する

どんな教材？

DBについて何かを理解し、DBを利用してPythonで出力する教材

目次

① DBについて理解しよう

②DBの設定をしよう

③Pythonでデータの出力をしよう

目次

① DBについて理解しよう

②DBの設定をしよう

③Pythonでデータの出力をしよう

① DBについて理解しよう

このページではDBについて何かを理解することができるよ

The screenshot shows the Amazon.co.jp search results for the keyword "本". The search results are displayed on a dark blue header with the Amazon logo and navigation links. Below the header, the search results are organized into a grid. On the left, there is a sidebar with filters for "無料配送の対象です" (Eligible for free delivery), "お届け日" (Delivery date), "カテゴリー" (Category), and "Prime Reading". The main content area shows a sponsored result for "社長の成功哲学" (The Success Philosophy of a CEO) by 日本経営合理化協会 出版局, followed by several other book titles including "成功の実現", "【新装版】人薄し術 (アラヤ鑑シリース)", and "地上最強の商人". Below the sponsored result, there is a "結果" (Results) section with four book covers: "人は話し方が9割", "戦略的いい人 残念ないい人の考え方", "7つの感情", and "デジタルの営業".

① DBについて理解しよう

DBはデータベース(data base)の略

データを保管したり、データを検索したりすることができるシステム

① DBについて理解しよう

例)小学校にDBがあるとしたら何が保管されているかを想像してみよう！

住所	学校名	生徒人数	講師数	教室数	面積
山口県	山口小	100	10	10	2000

質問

amazonのDBにはどのような列があるか考えてみよう

※考えてから次のスライドに進んでね

The screenshot shows the Amazon.co.jp search results for the keyword "本". The search bar at the top contains "すべて 本". The results section displays several book covers with their titles and ratings. The "結果" (Results) section at the bottom highlights four books:

- 「社長の成功哲学」 夢・希望・目標を実現し、事業や人生を成功に導く！ (著者: 日本経営合理化協会 出版局)
- 成功の実現 (★★★★☆ 310)
- 【新装版】人薄し術 (アラヤ謙シリース) (★★★★☆ 117)
- 地上最強の商人 (★★★★☆ 156)

The "結果" section also includes:

- 人は話し方が9割 (著者: 泉 健太郎)
- 戦略的いい人 残念ないい人の考え方 (著者: 小島 信)
- 7つの感情 (著者: 藤村 龍也)
- デジタルシブ 富士通式! (著者: 友田 邦彦)

答え

以下の表があるよ
他にもあるから自分で調べてみよう

商品ID	商品名	カテゴリー	価格	在庫数	説明	商品画像 URL	発売日	メーカー	レビュー

顧客ID	名前	メールアドレス	購入履歴	配送先住所	アカウント 作成日	支払い方法

① DBについて理解しよう

Amazonで検索するとこのDBからヒットするデータを見つけ出して、表示しているよ！

DBはほぼすべての企業や商品サイトで使われているから必ず概念を覚えておこう！

① DBについて理解しよう

DBの管理システムには様々なものがあり、
MySQL, PostgreSQLなどがあるよ

今回はxamppというアプリのMySQLを使ってDBの簡単な
設計やデータ抽出を行うよ

目次

① DBについて理解しよう

②DBの設定をしよう

③Pythonでデータの出力をしよう

②DBの設定をしよう

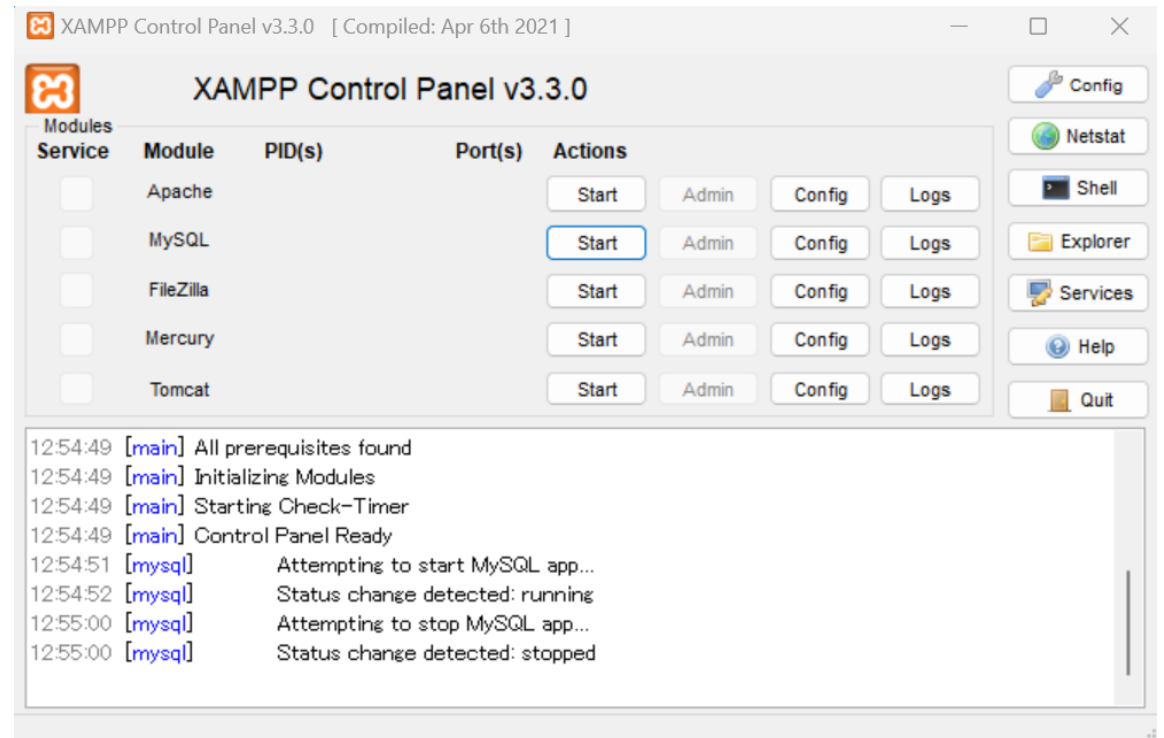
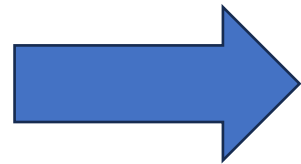
DBの設定をしてデータを格納できる準備ができるよ

The screenshot shows a database management interface with a menu bar at the top containing icons for '表示' (Display), '構造' (Structure), 'SQL', '検索' (Search), '挿入' (Insert), 'エクスポート' (Export), and 'インポート' (Import). Below the menu bar, a warning message states: '現在の選択には、ユニークなカラムが含まれていません。グリッド編集、チェックボックス、' (The current selection does not contain unique columns. Grid editing, checkboxes,). A green success message follows: '行 0 - 0 の表示 (合計 1, クエリの実行時間: 0.0005 秒。)' (Display 0 - 0 rows (Total 1, Query execution time: 0.0005 seconds.)). The SQL query 'SELECT * FROM `example`' is shown in a text area. Below the query, there are several options: 'プロファイリング [インライン編集] [編集] [EXPLAIN で確認] [PHP コードの作成] [更' (Profiling [Inline edit] [Edit] [Check with EXPLAIN] [Create PHP code] [More]). A control bar includes 'すべて表示' (Show all), '行数: 25' (Rows: 25), and '行フィルタ: このテーブルを検索' (Row filter: Search this table). An '拡張オプション' (Advanced options) button is also present. At the bottom, a table displays the query results:

ID	name	age
1	山田太郎	23

②DBの設定をしよう

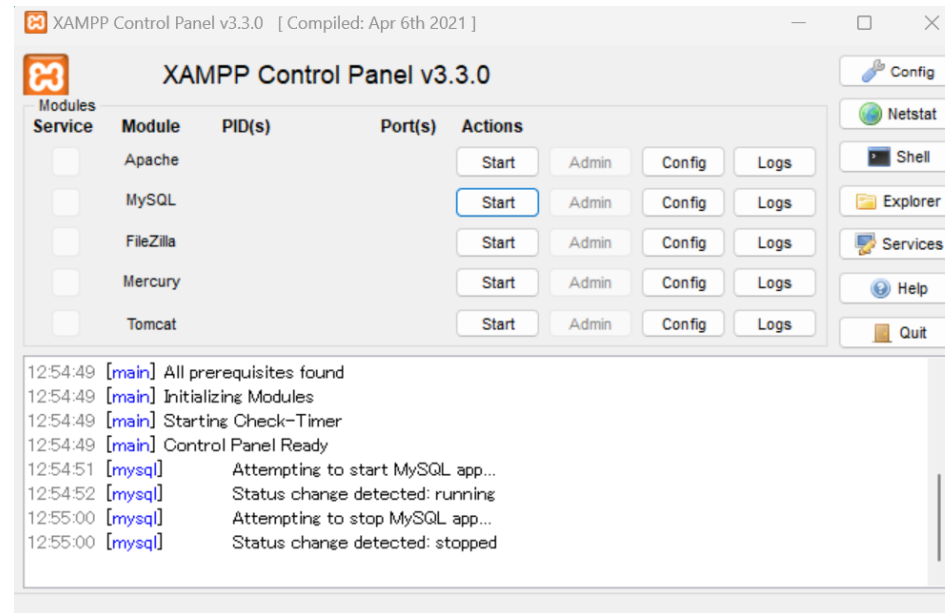
「xampp-control」をクリックすると右のような画面が開いたかな？



②DBの設定をしよう

左のボタンから「起動」「詳細」「設定」「ログ」になるよ

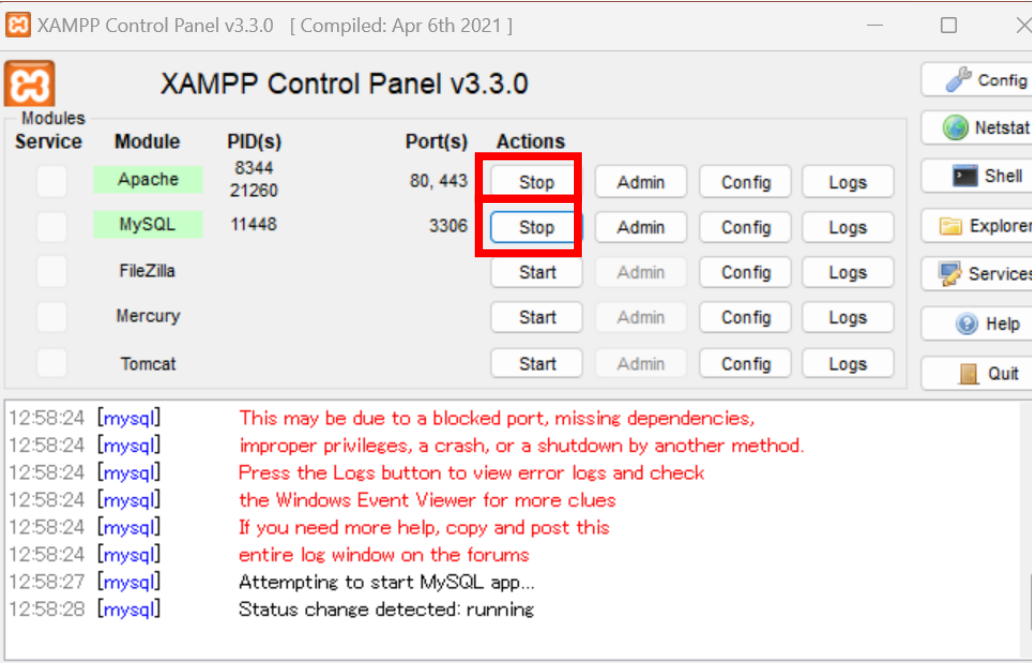
右の縦に並んでいるボタンは全体の設定や操作などのボタンだよ！



②DBの設定をしよう

上2つのシステムの「start」ボタンを押そう！

起動中は「stop」ボタンに変わるよ



XAMPP Control Panel v3.3.0 [Compiled: Apr 6th 2021]

XAMPP Control Panel v3.3.0

Service	Module	PID(s)	Port(s)	Actions
<input type="checkbox"/>	Apache	8344 21260	80, 443	Stop Admin Config Logs
<input type="checkbox"/>	MySQL	11448	3306	Stop Admin Config Logs
<input type="checkbox"/>	FileZilla			Start Admin Config Logs
<input type="checkbox"/>	Mercury			Start Admin Config Logs
<input type="checkbox"/>	Tomcat			Start Admin Config Logs

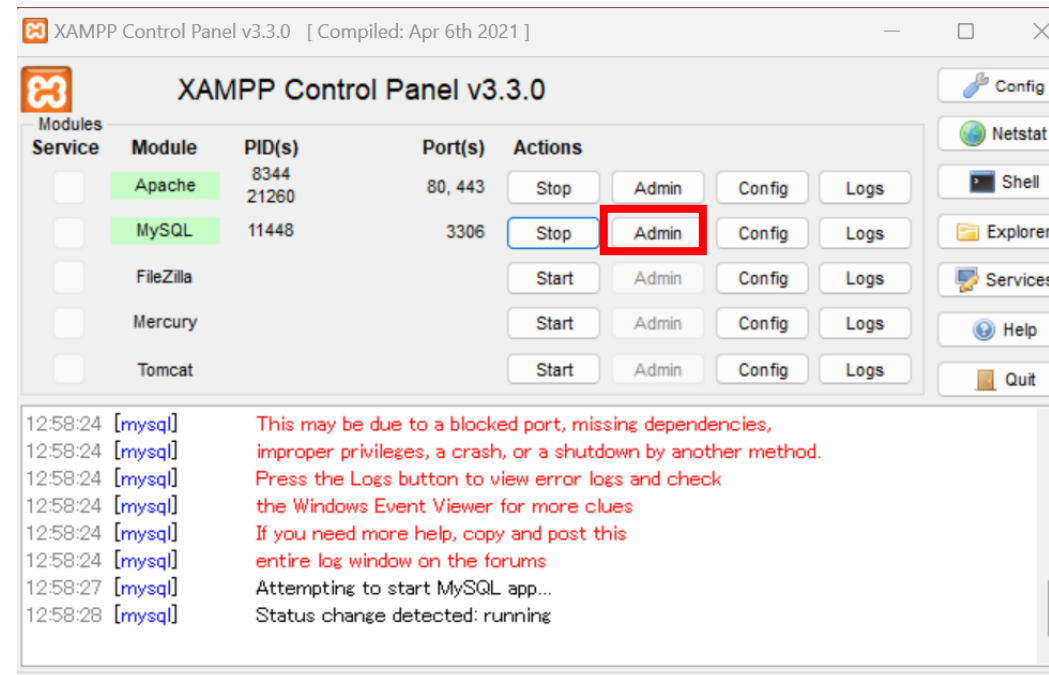
12:58:24 [mysql] This may be due to a blocked port, missing dependencies, improper privileges, a crash, or a shutdown by another method. Press the Logs button to view error logs and check the Windows Event Viewer for more clues. If you need more help, copy and post this entire log window on the forums

12:58:27 [mysql] Attempting to start MySQL app...

12:58:28 [mysql] Status change detected: running

②DBの設定をしよう

MySQLの「Admin」を押そう



②DBの設定をしよう

このような画面が出てきたかな？

サーバ: 127.0.0.1

データベース | SQL | 状態 | ユーザアカウント | エクスポート | インポート | 設定 | レプリケーション | その他

一般設定

サーバ接続の照合順序: utf8mb4_unicode_ci

詳細設定

外観の設定

言語 (Language): 日本語 - Japanese

テーマ: pmahomme | すべて表示

データベースサーバ

- サーバ: 127.0.0.1 via TCP/IP
- サーバの種類: MariaDB
- サーバの接続: SSLは使用されていません
- サーバのバージョン: 10.4.24-MariaDB - mariadb.org binary distribution
- プロトコルバージョン: 10
- ユーザ: root@localhost
- サーバの文字セット: UTF-8 Unicode (utf8mb4)

ウェブサーバ

- Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12
- データベースクライアントのバージョン: libmysql - mysqlnd 8.2.12
- PHP 拡張: mysqli | curl | mbstring
- PHP のバージョン: 8.2.12

phpMyAdmin

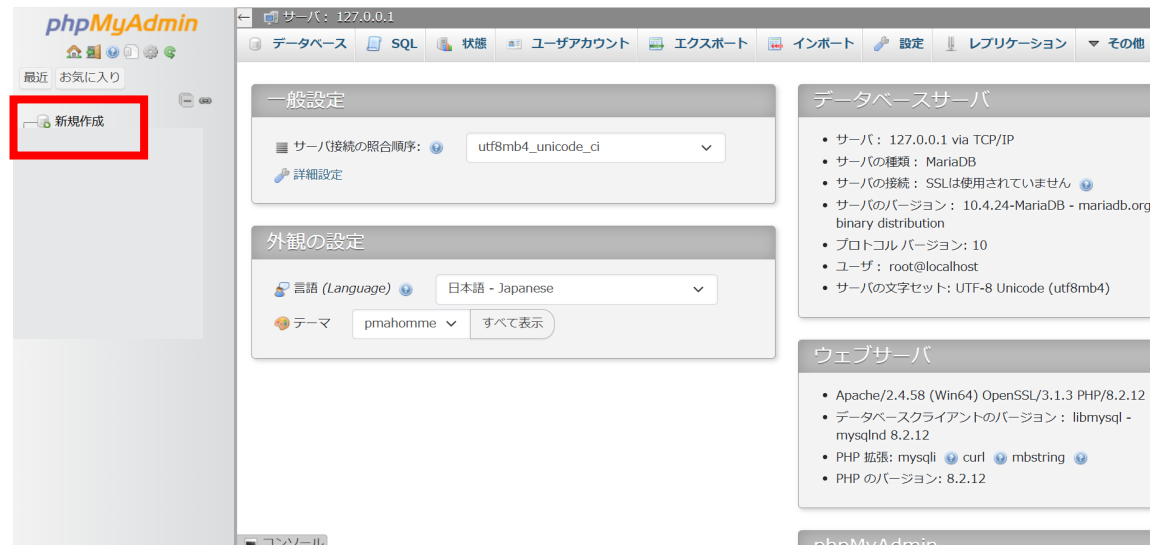
②DBの設定をしよう

次にテーブルを作ろう！

簡単に言えば表の枠組みを思ったらいいよ！

②DBの設定をしよう

左の「新規作成」を押そう

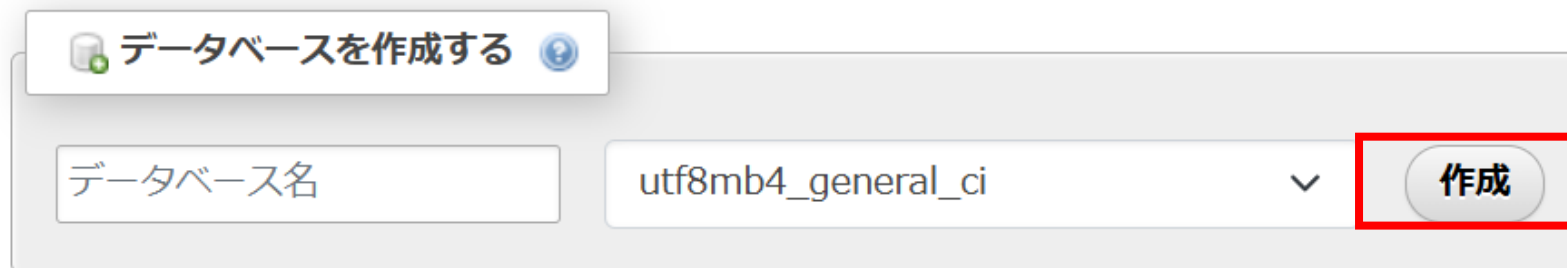


②DBの設定をしよう

データベースの名前を決めよう！

今回は「python_test」にするよ！

名前を書いたら「作成」ボタンを押そう



データベースを作成する

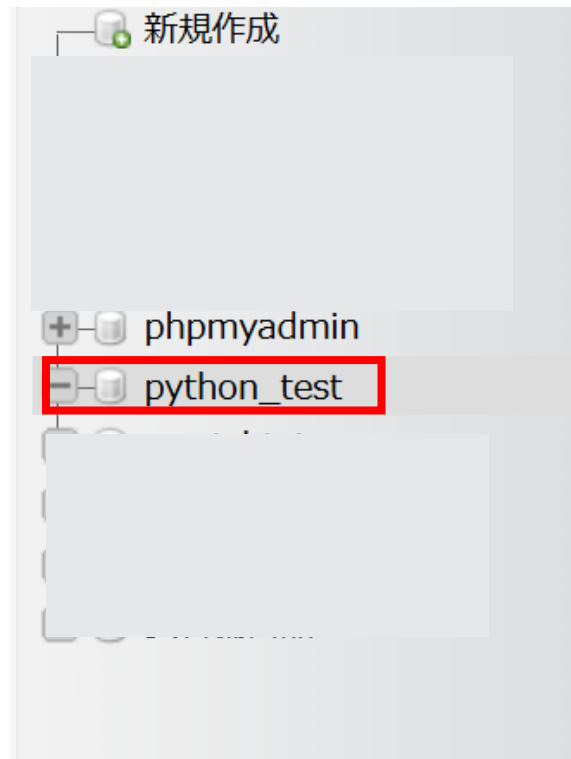
データベース名

utf8mb4_general_ci

作成

②DBの設定をしよう

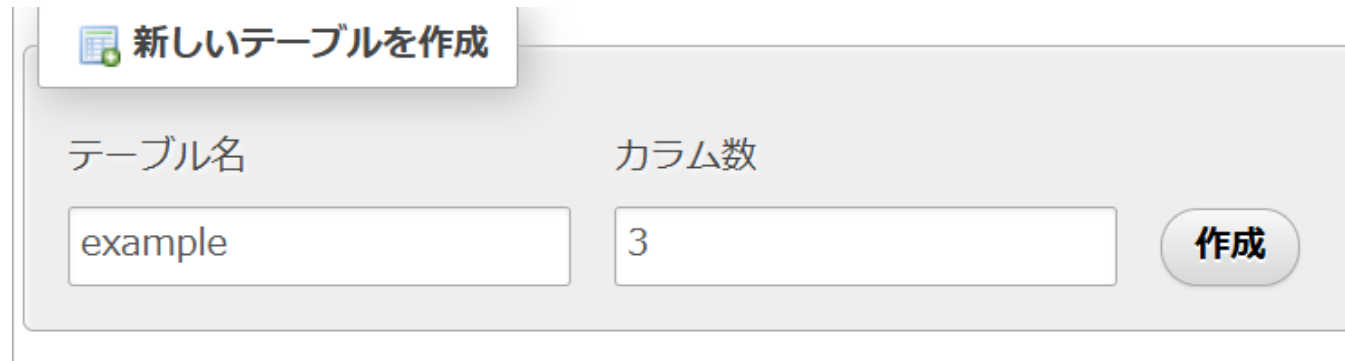
画面左側に新しいDBができたかな？



②DBの設定をしよう

次にテーブルの名前を決めようテキストでは「example」にしたよ！

カラムは3つにしよう



The image shows a screenshot of a web-based database creation interface. At the top, there is a title bar with a document icon and the text "新しいテーブルを作成" (Create new table). Below this, there are two input fields: "テーブル名" (Table name) containing the text "example" and "カラム数" (Number of columns) containing the number "3". To the right of these fields is a rounded button labeled "作成" (Create).

テーブル名	カラム数	
example	3	作成

②DBの設定をしよう

下の方に行くと保存ボタンがあるからボタンを押そう！

テーブルのコメント：

PARTITION 定義: 

パーティションによって:

パーティション:

②DBの設定をしよう

このようにさっき入力した内容が表形式で表示されたらOK
もし表示されなかったら「構造」のボタンを押して確認してみよう！



The screenshot shows a database management interface with a toolbar at the top. The '構造' (Structure) button is highlighted with a red box. Below the toolbar, there are two tabs: 'テーブルの構造' (Table Structure) and 'リレーションビュー' (Relation View). The 'テーブルの構造' tab is active, displaying a table with the following columns: #, 名前 (Name), タイプ (Type), 照合順序 (Collation), 属性 (Attributes), Null, デフォルト値 (Default Value), コメント (Comment), その他 (Other), and 操作 (Action). The table contains three rows of data:

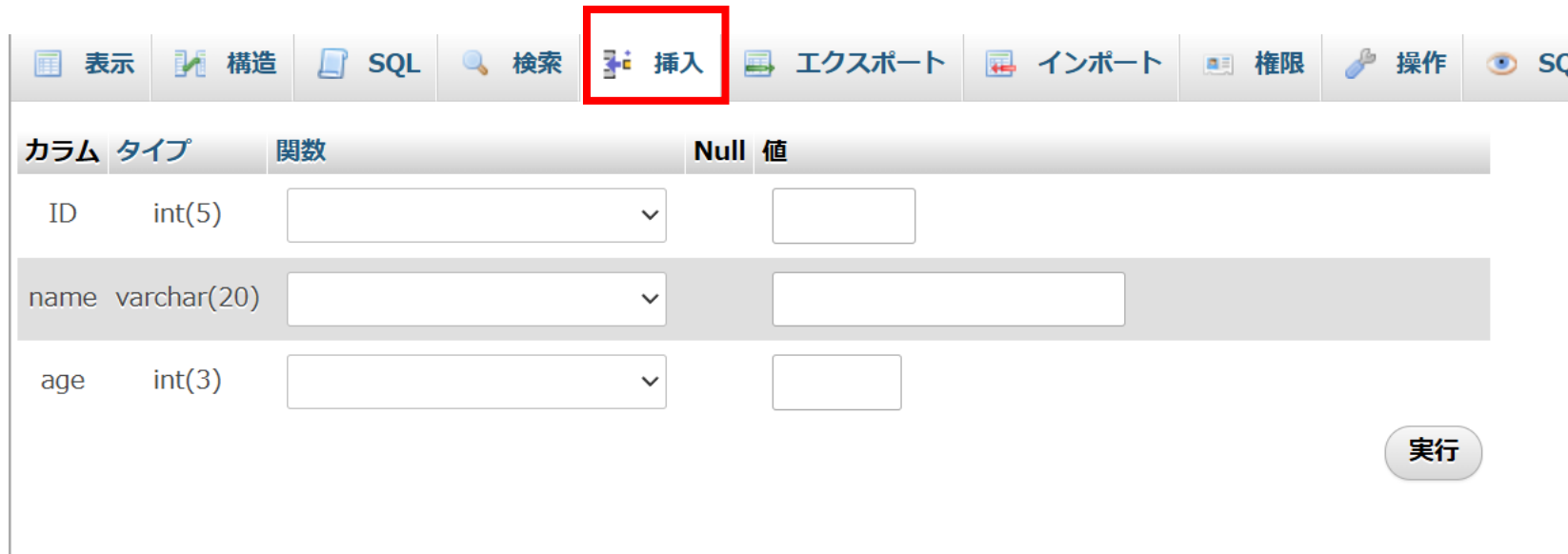
#	名前	タイプ	照合順序	属性	Null	デフォルト値	コメント	その他	操作
<input type="checkbox"/>	1 ID	int(5)			いいえ	なし			変更 削除 その他
<input type="checkbox"/>	2 name	varchar(20)	utf8mb4_general_ci		いいえ	なし			変更 削除 その他
<input type="checkbox"/>	3 age	int(3)			いいえ	なし			変更 削除 その他

At the bottom of the interface, there is a navigation bar with the following elements: an upward arrow, a checkbox labeled 'すべてチェックする' (Check all), the text 'チェックしたものを:' (Checked items:), and several icons for '表示' (Display), '変更' (Change), '削除' (Delete), '主' (Primary), and 'ユーーク' (User).

②DBの設定をしよう

最初に何か1つデータを手動で入力してみよう！

まずは「挿入」ボタンを押そう



The screenshot shows a database management interface with a toolbar at the top. The '挿入' (Insert) button is highlighted with a red box. Below the toolbar is a table with columns for 'カラム' (Column), 'タイプ' (Type), '関数' (Function), 'Null', and '値' (Value). The table has three rows: 'ID' (int(5)), 'name' (varchar(20)), and 'age' (int(3)). Each row has a dropdown menu for the function and a text input field for the value. An '実行' (Execute) button is located at the bottom right of the table.

カラム	タイプ	関数	Null	値
ID	int(5)	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
name	varchar(20)	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
age	int(3)	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>

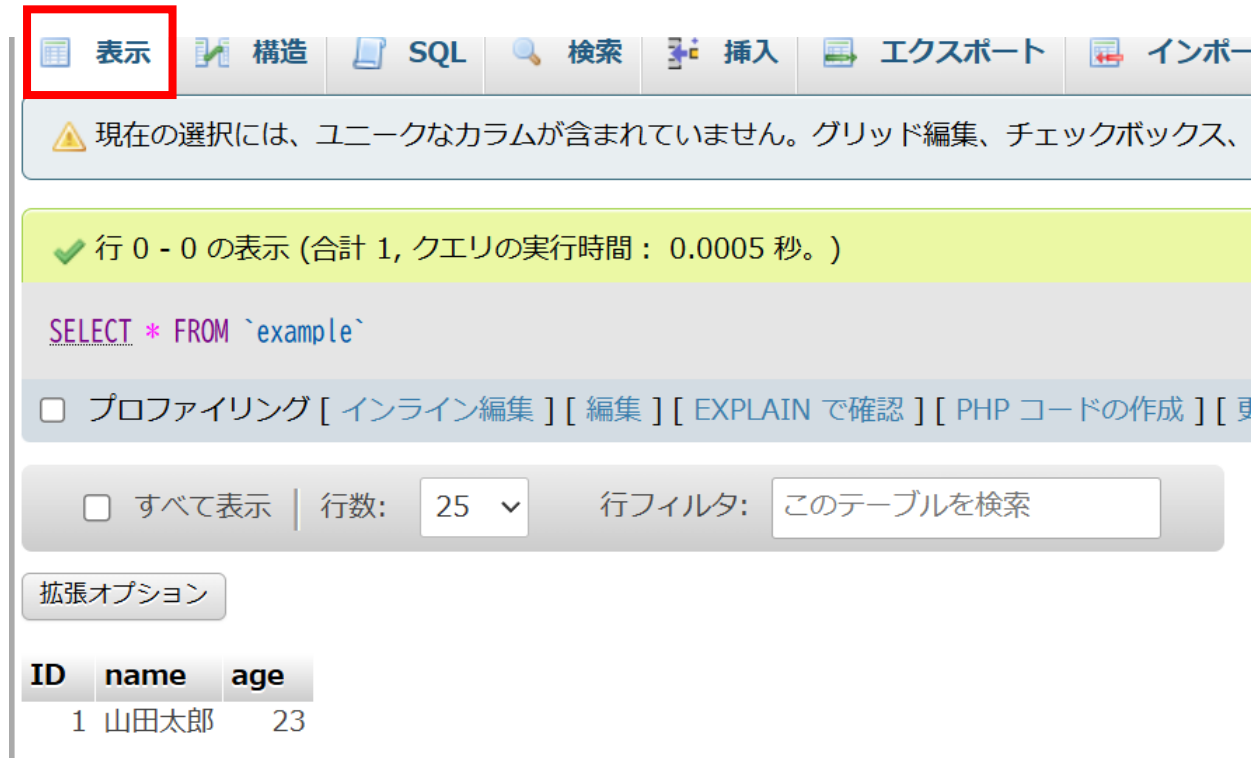
②DBの設定をしよう

ID、name、ageを適当に入力して実行を押してみよう

カラム	タイプ	関数	Null	値
ID	int(5)	<input type="text"/>	<input type="checkbox"/>	<input type="text" value="1"/>
name	varchar(20)	<input type="text"/>	<input type="checkbox"/>	<input type="text" value="山田太郎"/>
age	int(3)	<input type="text"/>	<input type="checkbox"/>	<input type="text" value="23"/>

②DBの設定をしよう

上の「表示」ボタンを押すとさっき入力した内容の表が下に表示されていることが確認できるかな？



表示 構造 SQL 検索 挿入 エクスポート インポート

⚠ 現在の選択には、ユニークなカラムが含まれていません。グリッド編集、チェックボックス、

✓ 行 0 - 0 の表示 (合計 1, クエリの実行時間 : 0.0005 秒。)

```
SELECT * FROM `example`
```

プロファイリング [インライン編集] [編集] [EXPLAIN で確認] [PHP コードの作成] [更

すべて表示 | 行数: 25 ▾ 行フィルタ: このテーブルを検索

拡張オプション

ID	name	age
1	山田太郎	23

②DBの設定をしよう

これでDBの設定と準備は完了

次からPythonを使って、表示を試みよう

目次

① DBについて理解しよう

②DBの設定をしよう

③Pythonでデータの出力をしよう

③Pythonでデータの出力をしよう

このページではPythonを使ってデータの出力ができるよ

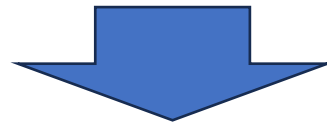
```
ID | Name | age
-----
1 | 山田太郎 | 23
```


③Pythonでデータの出力をしよう

自分で作ったフォルダーから環境を作ろう

今回は「DB_connect」にしたよ

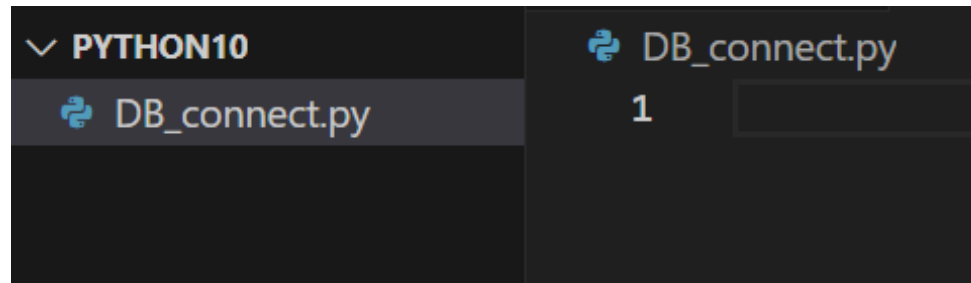
```
(base) PS C:\Users\小西 翔> cd .\python\  
(base) PS C:\Users\小西 翔\python> cd .\python10\  
(base) PS C:\Users\小西 翔\python\python10> ls  
(base) PS C:\Users\小西 翔\python\python10> conda create --name DB_connect python=3.9
```



```
(base) PS C:\Users\小西 翔\python\python10> conda activate DB_connect  
(DB_connect) PS C:\Users\小西 翔\python\python10>
```

③Pythonでデータの出力をしよう

VScodeを開いてPythonのファイルを作ろう



③Pythonでデータの出力をしよう

1行目にSQLに接続するコードを書こう

```
1 import pymysql
```

```
2
```

質問

ライブラリのインストールが必要だけどどのようなことをすればいいか考えてやってみよう

※考えてから次のスライドに進んでね

```
1 import pymysql
```

```
2
```

答え

ターミナル上でインストールすると使えるよ

```
(DB_connect) PS C:\Users\小西 翔\python\python10> pip install pymysql  
Collecting pymysql  
  Using cached PyMySQL-1.1.1-py3-none-any.whl.metadata (4.4 kB)  
Using cached PyMySQL-1.1.1-py3-none-any.whl (44 kB)  
Installing collected packages: pymysql  
Successfully installed pymysql-1.1.1
```

③Pythonでデータの出力をしよう

関数を作って、DBの接続のコードを書こう

```
3 def DB_example():
4     # MySQLに接続する
5     connection = pymysql.connect(
6         host='localhost',
7         user='root',
8         password='',
9         database='python_test',
10        cursorclass=pymysql.cursors.DictCursor # 結果を辞書形式で取得
11    )
```

説明

`pymysql.connect`: データベースに接続

`host`: 今回はローカルホストだが、実際はDBがあるIPアドレスなどを記載
`user`:
今回はroot

`password`: パスワードをDBに設定している場合はここにパスワードを記載

`database`: DBの名前を記載

```
3 def DB_example():
4     # MySQLに接続する
5     connection = pymysql.connect(
6         host='localhost',
7         user='root',
8         password='',
9         database='python_test',
10        cursorclass=pymysql.cursors.DictCursor # 結果を辞書形式で取得
11    )
```

③Pythonでデータの出力をしよう

DB内のテーブルからデータを獲得してみよう

```
12
13     try:
14         with connection.cursor() as cursor:
15             # データベースから情報を取得
16             cursor.execute("SELECT * FROM example") # 'example'テーブルから全てのデータを取得
17             DB_example = cursor.fetchall()
18
```


③Pythonでデータの出力をしよう

もしデータがある場合にはfor文を使ってデータの出力を試みよう。データがない場合はprint文でデータがないことを出力しよう

```
18
19     # データの表示
20     if DB_example:
21         print("ID | Name      | age")
22         print("-----")
23         for row in DB_example:
24             print(f"{row['ID']} | {row['name']:<10} | {row['age']}")
25     else:
26         print("No data found in the 'example' table.")
27
```

③Pythonでデータの出力をしよう

実行中にエラーが出た場合はエラー内容を入力するコードを書こう！

```
27  
28     except Exception as e:  
29         print(f"Error occurred: {e}")  
30
```

③Pythonでデータの出力をしよう

処理が終わったらDBとの接続を切ろう！

```
30  
31     finally:  
32         # 接続を閉じる  
33         connection.close()
```

③Pythonでデータの出力をしよう

最後に関数を書いてPythonを実行したときに関数が実行されるようにしよう

③Pythonでデータの出力をしよう

最後に関数実行されるコードを書いてPythonが実行されたときに関数が動くようにしよう

```
(DB_connect) PS C:\Users\小西 翔\python\python10> python .\DB_connect.py
ID | Name      | age
-----
1  | 山田太郎  | 23
```

お疲れさまでした

テキストは終了です。
あとは自分なりにアレンジを付け加えていこう！