

情報リテラシー（第6回 後期）

データベース

今日のねらい

- データベースの基本的な概念を理解できる
- SQLの基本的な構文（SELECT文）を習得できる
- オンラインツールを使って実際にSQLを実行できる
- データの検索・抽出・並べ替えができるようになる

データベースとは

データベース（Database）の定義

- **大量のデータを整理して保存**するシステム
- **効率的な検索・更新・管理**が可能
- 複数の人が同時にアクセスできる

身近なデータベースの例

- **Webサービス**： ユーザー情報、投稿データ
- **EC サイト**： 商品情報、注文履歴
- **学校**： 学生情報、成績データ
- **図書館**： 蔵書情報、貸出記録

なぜデータベースが必要？

- データの一元管理でミスを防ぐ
- 高速な検索が可能
- データの整合性を保つ

データベースの種類

主なデータベースの種類

1. リレーショナルデータベース (RDB)

- **表形式**でデータを管理
- 最も広く使われている
- 例：MySQL, PostgreSQL, SQLite, Oracle

2. NoSQLデータベース

- 表形式以外の構造
- 大規模・高速処理向け
- 例：MongoDB, Redis

→ **今日は最も基本的なリレーショナルデータベースを学びます**

DBMSとは

DBMS (Database Management System)

- **データベース管理システム**
- データベースを作成・管理・操作するためのソフトウェア
- ユーザーやアプリケーションとデータベースの間に立つ

DBMSの役割

- **データの保存と管理**： 効率的にデータを格納
- **データの検索**： 高速な検索処理
- **データの整合性**： 矛盾のないデータを維持
- **同時アクセス制御**： 複数ユーザーの同時利用
- **セキュリティ**： アクセス権限の管理

代表的なDBMS

主なDBMS製品

オープンソース（無料）

- **MySQL**：最も普及しているオープンソースDBMS
- **PostgreSQL**：高機能で信頼性が高い
- **SQLite**：軽量で組み込み型、今日使用

商用製品

- **Oracle Database**：大企業向け、高性能
- **Microsoft SQL Server**：Windows環境で人気
- **IBM Db2**：大規模システム向け

使い分け

- **小規模・学習用**：SQLite
- **Web アプリケーション**：MySQL, PostgreSQL
- **大規模業務システム**：Oracle, SQL Server

リレーショナルデータベースの基本

基本的な構造

テーブル (Table)

- データを格納する **表**
- Excelのシートに似ている

レコード (Record) / 行 (Row)

- 1件分のデータ
- 例：1つの商品情報

フィールド (Field) / 列 (Column)

- データの項目
- 例：商品名、価格、在庫数

テーブルの例：商品テーブル

product_id	name	price	category	stock
1	ノートPC	89800	電子機器	15
2	マウス	1980	周辺機器	50
3	キーボード	5980	周辺機器	30
4	モニター	24800	電子機器	20

- **テーブル名**： products（商品テーブル）
- **レコード数**： 4件
- **フィールド**： product_id, name, price, category, stock

主キーとは

主キー（Primary Key）

- 各レコードを一意に識別するためのフィールド
- 重複しない値を持つ
- 通常は `id` や `oo_id` という名前

商品テーブルの主キー

`product_id` → 各商品を識別する一意の番号

主キーの役割

- データの重複を防ぐ
- レコードを確実に特定できる
- 他のテーブルとの関連付けに使う

SQLとは

SQL (Structured Query Language)

- データベースを**操作するための言語**
- 「エスキューエル」または「シークェル」と読む
- **標準化された言語**なので、多くのデータベースで使える

SQLでできること

- **SELECT** : データの検索・取得
- **INSERT** : 新しいデータの追加
- **UPDATE** : データの更新
- **DELETE** : データの削除

→ **今日はSELECT文（データの検索）を学びます**

SELECT文の基本

SELECT文の構文

```
SELECT 列名 FROM テーブル名;
```

すべての列を取得

```
SELECT * FROM products;
```

- *****（アスタリスク）はすべての列を意味する

特定の列だけを取得

```
SELECT name, price FROM products;
```

- 商品名と価格だけを取得

演習1：SQLite Onlineの使い方

準備：SQLite Onlineを開く

1. ブラウザで <https://sqliteonline.com/> にアクセス
2. 左側にSQLエディタが表示される

テーブルを作成する

左側のSQLエディタに以下を入力して「**Run**」をクリック

```
DROP TABLE IF EXISTS products;

CREATE TABLE products (
  product_id INTEGER PRIMARY KEY,
  name TEXT,
  price INTEGER,
  category TEXT,
  stock INTEGER
);
```

演習1：データを挿入する

データを挿入

続けて以下のSQL文を入力して「Run」をクリック

```
INSERT INTO products VALUES (1, 'ノートPC', 89800, '電子機器', 15);  
INSERT INTO products VALUES (2, 'マウス', 1980, '周辺機器', 50);  
INSERT INTO products VALUES (3, 'キーボード', 5980, '周辺機器', 30);  
INSERT INTO products VALUES (4, 'モニター', 24800, '電子機器', 20);
```

確認

```
SELECT * FROM products;
```

全データが表示されればOK！

演習2：SELECT文の基本

各課題を1つずつ入力して「Run」をクリックしよう！

課題1：すべてのデータを表示

```
SELECT * FROM products;
```

課題2：商品名と価格だけを表示

```
SELECT name, price FROM products;
```

課題3：商品名、カテゴリ、在庫を表示

```
SELECT name, category, stock FROM products;
```

WHERE句：条件を指定してデータを抽出

WHERE句の構文

```
SELECT 列名 FROM テーブル名 WHERE 条件;
```

例1：価格が10000円以上の商品

```
SELECT * FROM products WHERE price >= 10000;
```

例2：カテゴリが「周辺機器」の商品

```
SELECT * FROM products WHERE category = '周辺機器';
```

例3：在庫が20個以下の商品

```
SELECT name, stock FROM products WHERE stock <= 20;
```

ORDER BY：並べ替え

ORDER BY句の構文

```
SELECT 列名 FROM テーブル名 ORDER BY 列名;
```

昇順（小さい順）

```
SELECT * FROM products ORDER BY price;
```

→ 価格の安い順に表示

降順（大きい順）

```
SELECT * FROM products ORDER BY price DESC;
```

→ 価格の高い順に表示（DESCをつける）

LIMIT：件数を制限

LIMIT句の構文

```
SELECT 列名 FROM テーブル名 LIMIT 件数;
```

上位3件を取得

```
SELECT * FROM products ORDER BY price DESC LIMIT 3;
```

→ 価格が高い順に3件だけ表示

WHERE、ORDER BY、LIMIT の組み合わせ

```
SELECT name, price FROM products  
WHERE category = '周辺機器'  
ORDER BY price  
LIMIT 2;
```

→ 周辺機器を価格の安い順に2件表示

演習3：SELECT文の実践

以下のSQL文を作成して実行してみよう！
各課題を1つずつ入力して「Run」をクリック！

課題1

価格が5000円以上の商品の商品名と価格を表示せよ

課題2

電子機器のカテゴリで、価格の順に並べて表示せよ

演習3の解答例

課題1の解答

```
SELECT name, price FROM products WHERE price >= 5000;
```

課題2の解答

```
SELECT * FROM products  
WHERE category = '電子機器'  
ORDER BY price;
```

集計関数の紹介

よく使う集計関数

SQLには便利な集計関数がある（今日は紹介のみ）

- **COUNT()**：レコード数を数える

```
SELECT COUNT(*) FROM products;
```

- **SUM()**：合計を計算

```
SELECT SUM(stock) FROM products;
```

- **AVG()**：平均を計算

```
SELECT AVG(price) FROM products;
```

- **MAX()**、**MIN()**：最大値、最小値

```
SELECT MAX(price), MIN(price) FROM products;
```

まとめ

- **データベース**は大量データを効率的に管理するシステム
- **リレーショナルデータベース**は表形式でデータを管理
- **SQL**はデータベースを操作する標準言語
- **SELECT文**でデータの検索・抽出ができる
 - WHERE：条件指定
 - ORDER BY：並べ替え
 - LIMIT：件数制限
- データベースは現代社会のあらゆる場面で使われている

今日学んだSQLの基本を活用しよう！

感想をチャット欄に書こう！