

## 14章 関数を使ってデータを集計しよう

---

# 14章 関数を使ってデータを集計しよう

SQLの関数を使ってレコード数やデータの合計値、平均値などを取得する方法を学びます。

## 本章の目標

- レコード数を取得する方法を知ること
- データの合計値を取得する方法を知ること
- データの平均値を取得する方法を知ること
- データの最大値と最小値を取得する方法を知ること

# 14章 これまでの学習内容

これまでではSQLのSELECT文を使って単純にデータそのものを取得してきました。しかし、SQLの「集計関数」を使えば、データ数やデータの合計、平均、最大値、最小値などを取得できます。

「1日あたりの売上や平均を知りたい」「月間のユーザー登録者数を知りたい」など、データを分析する場面ではとても役立つので、本章を進めてぜひ覚えておきましょう。

# 14章 COUNT関数でレコード数を取得しよう

アプリやサービスでは、例えば「都道府県別のユーザー数を知りたい」「全体のユーザー登録者数を知りたい」など、条件に一致するレコード数をカウントしたい場面が出てきます。

そこで使うのが**COUNT関数**です。

**COUNT**は「数を数える」という意味で、条件に一致するレコード数を取得するSQLの関数です。

# 14章 COUNT関数の使い方

COUNT関数の基本的な使い方は以下のとおりです。

```
-- テーブルのレコード数を取得する  
SELECT COUNT(*) FROM テーブル名;
```

**これまではSELECTのあとに取得したいカラム名を指定していましたが、COUNT関数をはじめとする集計関数を使う場合、集計関数そのものを指定します。**

なお、集計関数の引数には集計対象のカラム名を指定するのですが、COUNT関数の場合は `*`（アスタリスク）を指定することが多いです。

# 14章 COUNT関数の引数について

カラム名を指定した場合は、指定したカラムについて「NULL以外のレコードが何件あるか」をカウントできます。この場合、NULLが保存されているレコードはカウントされません。

そのため、「COUNT(\*)でカウントしたら100件取得できたのに、COUNT(address)を指定したら98件しか取得できなかった」ということもありえます。この例では、指定したカラム（address）に2件のNULLが存在することを意味します。

## まとめ

- **COUNT(\*)** : NULLかどうかにかかわらず、すべてのレコード数を取得する
- **COUNT(カラム名)** : NULLを除いたレコード数を取得する

## 14章 COUNT関数を使ってみよう

では実際に、COUNT関数を使ってみましょう。今回はusersテーブルのレコード数を取得してみます。

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

### 実行結果

SQL文の実行結果を見ると、usersテーブルのレコード数が30件であることがわかります。

COUNT( * )
30

# 14章 SUM関数で合計値を取得しよう

アプリやサービスでは、例えば「1日あたりの売上を知りたい」「ユーザー1人あたりの課金額を知りたい」など、データの合計値を取得したい場面が出てきます。

そこで使うのが**SUM関数**です。

**SUM**は「～を合計する」という意味で、指定したカラムの合計値を取得するSQLの関数です。



# 14章 SUM関数の使い方

SUM関数の基本的な使い方は以下のとおりです。

-- 指定したカラムの合計値を取得する  
**SELECT SUM**(カラム名) **FROM** テーブル名;

## 14章 SUM関数を使ってみよう

では実際に、SUM関数を使ってみましょう。今回はusersテーブルのageカラムを指定して、ユーザーの総年齢を取得してみます。

```
SELECT SUM(age) FROM users;
```

### 実行結果

SQL文の実行結果を見ると、ユーザーの総年齢が1468歳であることがわかります。

SUM(age)
1468

# 14章 AVG関数で平均値を取得しよう

アプリやサービスでは、例えば「1日の平均売上を知りたい」「ユーザーの平均年齢を知りたい」など、データの平均値を取得したい場面が出てきます。

そこで使うのが**AVG関数**です。

**AVG**は「～の平均値を求める」を意味するaverageの略で、指定したカラムの平均値を取得するSQLの関数です。

# 14章 AVG関数の使い方

AVG関数の基本的な使い方は以下のとおりです。

-- 指定したカラムの平均値を取得する  
**SELECT** **AVG**(カラム名) **FROM** テーブル名;

## 14章 AVG関数を使ってみよう

では実際に、AVG関数を使ってみましょう。今回はusersテーブルのageカラムを指定して、ユーザーの平均年齢を取得してみます。

```
SELECT AVG(age) FROM users;
```

### 実行結果

SQL文の実行結果を見ると、ユーザーの平均年齢が約49歳であることがわかります。

AVG( age )
48.9333

# 14章 MAX関数とMIN関数で最大値と最小値を取得しよう

アプリやサービスでは、例えば「最も売れた商品の個数を知りたい」「最年少のユーザーの年齢を知りたい」など、データの最大値や最小値を取得したい場面が出てきます。

そこで使うのが**MAX関数**と**MIN関数**です。

- **MAX** : 「最大の」を意味するmaximumの略
- **MIN** : 「最小の」を意味するminimumの略

MAXは指定したカラムの最大値を取得するSQLの関数、MINは最小値を取得するSQLの関数です。

# 14章 MAX関数とMIN関数の特徴

**なお、MAX関数とMIN関数は数値型のデータだけでなく、文字列型や日付型のデータの最大値や最小値も取得できます。**

文字列型のデータであれば7章で学んだ照合順序、日付型であれば日付の新旧（新＝大、旧＝小）で大小を判定します。

# 14章 MAX関数とMIN関数の使い方

MAX関数とMIN関数の基本的な使い方は以下のとおりです。

-- 指定したカラムの最大値を取得する  
**SELECT MAX**(カラム名) **FROM** テーブル名;

-- 指定したカラムの最小値を取得する  
**SELECT MIN**(カラム名) **FROM** テーブル名;



## 14章 MAX関数を使ってみよう

では実際に、MAX関数を使ってみましょう。MAX関数の引数にageカラムを指定して、最高齢のユーザーを取得してみます。

```
SELECT MAX(age) FROM users;
```

### 実行結果

SQL文の実行結果を見ると、ユーザーの最高齢が77歳であることがわかります。

MAX( age )
77

## 14章 MIN関数を使ってみよう

続いてMIN関数の引数にageカラムを指定して、最年少のユーザーを取得してみましょう。

```
SELECT MIN(age) FROM users;
```

### 実行結果

SQL文の実行結果を見ると、ユーザーの最年少が21歳であることがわかります。

MIN( age )
21

# 14章 まとめ (1/3)

- SQLの集計関数を使うことで、データそのものではなくデータ数やデータの合計、平均、最大値、最小値などを取得できる
- **COUNT関数**：テーブルのレコード数を取得する
- **SUM関数**：指定したカラムの合計値を取得する
- **AVG関数**：指定したカラムの平均値を取得する
- **MAX関数／MIN関数**：指定したカラムの最大値／最小値を取得する

# 14章 まとめ (2/3)

- **COUNT**は「数を数える」という意味で、条件に一致するレコード数を取得するSQLの関数である
  - **COUNT(\*)**：NULLかどうかにかかわらず、すべてのレコード数を取得する
  - **COUNT(カラム名)**：NULLを除いたレコード数を取得する
- **SUM**は「～を合計する」という意味で、指定したカラムの合計値を取得するSQLの関数である
- **AVG**は「～の平均値を求める」を意味するaverageの略で、指定したカラムの平均値を取得するSQLの関数である
- **MAX**は「最大の」を意味するmaximumの略で、指定したカラムの最大値を取得するSQLの関数である
- **MIN**は「最小の」を意味するminimumの略で、指定したカラムの最小値を取得するSQLの関数である

# 14章 まとめ (3/3)

-- テーブルのレコード数を取得する

```
SELECT COUNT(*) FROM テーブル名;
```

-- 指定したカラムの合計値を取得する

```
SELECT SUM(カラム名) FROM テーブル名;
```

-- 指定したカラムの平均値を取得する

```
SELECT AVG(カラム名) FROM テーブル名;
```

-- 指定したカラムの最大値を取得する

```
SELECT MAX(カラム名) FROM テーブル名;
```

-- 指定したカラムの最小値を取得する

```
SELECT MIN(カラム名) FROM テーブル名;
```