

# 第2回：列を選ぶ - SELECT 列名

---

必要な情報だけを抜き出そう！

# ● 今日のゴール

---

- ✓ 巨大なデータから、自分が見たい列だけを抜き出せるようになる

# ● 座学パート

---

## 1. なぜ列を選ぶ必要があるの？

実際のデータベースには**数十～数百の列**があることも。

全部見ていたら画面に収まらないし、必要な情報が埋もれてしまいます。

## 2. SELECT文の構文

-- 基本形

```
SELECT 列名1, 列名2 FROM 'ファイル名';
```

-- 複数の列を選ぶ

```
SELECT customer_id, product_id, quantity FROM 'data/sales.csv';
```

-- 列に別名をつける (AS)

```
SELECT customer_name AS お名前 FROM 'data/customers.csv';
```

## 3. ポイント

列名は**カンマ（,）**で区切る

列名の順番は自由に変えられる

**AS** を使うと、結果の列名を変更できる（日本語もOK!）

## 4. DuckDBでの操作メリット

結果がすぐに表形式で確認できる

列名が見やすく表示される

エラーメッセージが分かりやすい

# ● 演習パート

---

## 演習1：sales.csvから「商品ID」と「数量」だけを表示

クエリ入力エリアに入力：

```
SELECT product_id, quantity FROM 'data/sales.csv';
```

実行ボタンをクリック（または Ctrl/Cmd + Enter）

# 結果を確認

**2列だけ**が表示される

全15行のデータが見える

他の列（customer\_id, order\_date）は表示されない



## 演習2：products.csvから「商品名」と「価格」を表示

```
SELECT product_name, price FROM 'data/products.csv';
```



### 確認ポイント：

商品名が日本語で正しく表示されている

価格が数値として表示されている

文字化けもすぐに気づける

## 演習3：customers.csvの「顧客名」を「お名前」という列名で表示

まずはシンプルに：

```
SELECT customer_name AS お名前 FROM 'data/customers.csv';
```

# より実用的に

```
SELECT
  customer_id AS 顧客コード,
  customer_name AS お名前,
  email AS メールアドレス
FROM 'data/customers.csv';
```



## 便利機能：

改行して見やすく書ける

結果の列名が日本語で表示される

列幅を調整して見やすくできる

# ● 応用練習

---

## 1. 列の順番を変えてみよう

```
-- 元の順番と違う順番で表示
SELECT order_date, customer_id, quantity
FROM 'data/sales.csv'
LIMIT 5;
```

## 2. 同じ列を2回表示してみよう

```
SELECT price, product_name, price  
FROM 'data/products.csv';
```

→ priceが2回表示されることを確認

### 3. 見やすいレポートを作ろう

```
SELECT
    product_name AS 商品名,
    price AS 単価,
    category AS カテゴリー
FROM 'data/products.csv'
ORDER BY price DESC;
```

(ORDER BYは第4回で詳しく学びます)

# ● 実践的な使い方

---

## ケース1：売上日報の作成

```
-- 売上日報に必要な情報だけ
SELECT
    order_date AS 売上日,
    product_id AS 商品コード,
    quantity AS 販売数
FROM 'data/sales.csv'
LIMIT 10;
```

## ケース2：顧客メールリストの作成

```
-- メール送信用リスト
SELECT
    customer_name AS 氏名,
    email AS 送信先アドレス
FROM 'data/customers.csv';
```



### 活用法：

結果をそのままコピーしてExcelに貼り付け可能

見た目を確認しながらクエリを調整できる



# ● デバッグ方法

## 列名を間違えた場合

-- わざと間違えてみる


```
SELECT customername FROM 'data/customers.csv';
```

⚠ エラーメッセージで「column "customername" does not exist」と表示される

→ 正しくは `customer\_name` (アンダースコアが必要)

## 存在しない列を指定した場合

```
SELECT customer_id, phone FROM 'data/customers.csv';
```

 phoneという列は存在しないというエラー

# ● 本日のまとめ

---

今日学んだこと：

- ✓ `SELECT 列名1, 列名2 FROM ...` で必要な列だけ選択
- ✓ 列名はカンマで区切る
- ✓ `AS` を使って列に別名をつけられる
- ✓ 結果がすぐに確認できて便利
- ✓ エラーも分かりやすく表示される

# よく使うパターン

```
-- パターン1：シンプルな列選択  
SELECT customer_id, customer_name  
FROM 'data/customers.csv';
```

```
-- パターン2：日本語の列名で見やすく  
SELECT  
    customer_id AS 顧客ID,  
    customer_name AS 顧客名  
FROM 'data/customers.csv';
```

# ● 次回予告

---

第3回では、WHERE句を使って特定の条件に合うデータだけを取り出す方法を学びます。

結果を見ながら、条件を調整していくことができます！

# ● 追加演習

---

## 問題1：必要な情報だけを抽出

以下の要件に従ってクエリを作成してください：

- 1. customers.csvから顧客IDとメールアドレスだけを表示
- あなたの答えをここに書いてください
  
- 2. products.csvから商品名だけを表示（列名は変更しない）
- あなたの答えをここに書いてください
  
- 3. sales.csvから注文日と数量だけを表示
- あなたの答えをここに書いてください

## 問題2：意味のある列名に変更

以下のクエリを完成させてください：

```
-- 顧客一覧を日本語で分かりやすく表示
SELECT
    customer_id AS ____, -- 「顧客番号」という列名にする
    customer_name AS ____, -- 「お客様名」という列名にする
    address AS ____ -- 「ご住所」という列名にする
FROM 'data/customers.csv';
```

## 問題2（続き）

```
-- 商品マスタを見やすく表示
SELECT
    product_name AS 商品名,
    price AS 販売価格,    -- priceを「販売価格」に
    category AS 商品分類  -- categoryを「商品分類」に
FROM 'data/products.csv';
```



## 問題3：列の順番を工夫

目的に応じて列の順番を変えてみましょう：

- 1. メール送信用リスト（メール、名前、IDの順）
- あなたの答えをここに書いてください
  
- 2. 価格表（価格、商品名、カテゴリの順）
- あなたの答えをここに書いてください

## 問題4：よくある間違いを修正

以下のクエリのエラーを修正してください：

-- エラー1：カンマ忘れ

```
SELECT customer_id customer_name FROM 'data/customers.csv';
```

-- エラー2：存在しない列名

```
SELECT id, name FROM 'data/customers.csv';
```

-- エラー3：ASの位置が間違い

```
SELECT customer_name お名前 AS FROM 'data/customers.csv';
```

## チャレンジ問題

- sales.csvから以下の形式でレポートを作成してください：
- 「2024年1月○日に顧客○○○が商品○○○を○個購入」
- ヒント：列名を工夫して、見た人が分かりやすいレポートにしましょう

## 実践問題：請求書フォーマット

- sales.csvとproducts.csvの知識を組み合わせる（JOINは使わずに）
- 請求書に必要な列だけを選んで、適切な日本語名をつけてください
- 例：取引日、商品コード、数量など

# ● FAQ

---

**Q: 列名の大文字小文字は区別されますか？**

A: DuckDBでは区別されません。 `customer_id` も `CUSTOMER_ID` も同じです。

**Q: AS は必須ですか？**

A: 省略可能ですが、読みやすさのためASを使うことを推奨します。

**Q: 結果をファイルに保存できますか？**

A: CSVエクスポートが可能です。または、結果を選択してコピー＆ペーストもできます。