

## 副問い合わせ

1日目

2日目

3日目

4日目

5日目

6日目

7日目

8日目

9日目

10日目

11日目

12日目

13日目

14日目

15日目

16日目

17日目

18日目

19日目

20日目

21日目

テーブルに別名をつける

# 別名をつける

これまでSQLでカラムに別名をつけて表示してきた。同じように、テーブルにも別名をつけることができる

## カラムに別名をつける

```
SELECT column_name AS alias_name  
FROM table_name;
```

## テーブルに別名をつける

```
SELECT  
  st.id,  
  st.name  
FROM  
  students AS st;
```

副問い合わせ

# 副問い合わせとは

SELECT文をFROMの中やWHEREの中などに記述して、SELECTの結果を別の処理に用いるSQL文

メインのSELECT文以外のSELECTをサブクエリ(副問い合わせ)と言う

メインのSELECT文をメインクエリ(主問い合わせ)と言う

SELECT文1

SELECT文2

SELECT文3

SELECT1はSELECT2を含み

SELECT2はSELECT3を含んでいる

# 副問い合わせの構文 1 (INで使う)

INの後に()で囲いその中にSELECTを記述すると、SELECTの結果に含まれるレコードだけ取り出すことができる。

```
SELECT
  lastName, firstName
FROM
  employees
WHERE
  office_code IN (SELECT
    office_code
  FROM
    offices
  WHERE
    country = 'USA');
```

SELECTでofficesテーブルから、countryがUSAの人のoffice\_codeだけ取り出す

employeesテーブルからoffice\_codeが、SELECTの結果に存在するものを取り出す

# 副問い合わせの構文2 (INで複数のカラムを使う)

INの後に()で囲いその中にSELECTを記述して複数のカラムを取得する。

```
SELECT
  lastName, firstName
FROM
  employees
WHERE
  (office_code, office_name) IN (SELECT
    office_code, office_name
  FROM
    offices
  WHERE
    country = 'USA');
```

office\_codeとoffice\_nameの行だけ取り出す

## 副問い合わせの構文3 (集計関数を使う)

WHERE句の比較式に、()で囲ったSELECT文を記述し副問い合わせで集計した値と比較する

```
SELECT
  customerNumber,
  checkNumber,
  amount
FROM
  payments
WHERE
  amount = (SELECT MAX(amount) FROM payments);
```

SELECT MAXでpaymentsテーブルから、amountの最大値を取り出す

paymentsテーブルからamountが、SELECTで取得したMAX(amount)に等しいものを取得する



# 副問い合わせの構文 4 (FROMの取得先に用いる)

FROMの取得対象のテーブルの代わりに、()でSELECTを記述する

```
SELECT
  MAX(lineitems.items),
  MIN(lineitems.items),
  FLOOR(AVG(lineitems.items))
FROM
  (SELECT
    order_number, COUNT(order_number) AS items
  FROM
    order_details
  GROUP BY orderNumber) AS lineitems;
```

order\_detailsテーブルから取得したデータをFROMで利用する

サブクエリで取得した表には、lineitemsという別名を付ける

# 副問い合わせの構文5 (SELECTの行の1つに用いる)

SELECTで取得する対象の行に含める

```
SELECT
  p1.site_name,
  (
    SELECT
      MAX(file_size)
    FROM
      pages2 AS p2
    WHERE
      p2.site_id = p1.site_id
  ) AS max_file_size
FROM
  pages1 AS p1;
```

FROM pages2でpages2テーブルからレコードを取得する。pages1のsite\_idに等しいsite\_idのレコードに絞り込んでfile\_sizeの**最大値**を求める

pages1には、別名でp1をつける

# 副問い合わせの構文6 (CASEとともに使う)

SELECTで取得する対象の行に含める

```
SELECT
  employee_id,
  last_name,
  (
    CASE
      WHEN department_id =(
        SELECT
          department_id
        from
          departments
        WHERE
          location_id = 2500
      ) THEN 'Canada'
      ELSE 'USA'
    END
  ) location
FROM
  employees
```

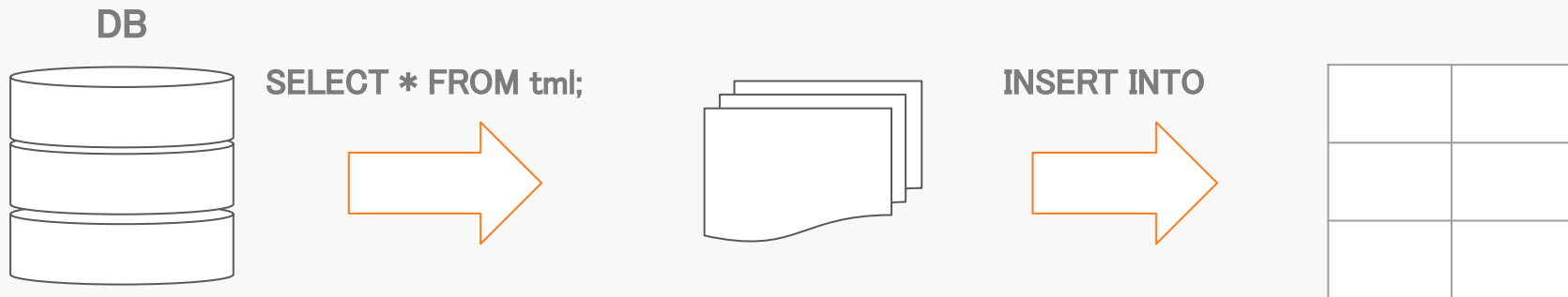
departmentsテーブルから、location\_idが2500の  
department\_idの値を取得する

employeesテーブルのdepartment\_idが、サブクエ  
リに等しい場合は「Canada」それ以外は「USA」と  
表示する

**INSERT INTO SELECT, CREATE TABLE SELECT**

# INSERT INTO SELECTとは

SELECT処理の実行結果をテーブルに挿入する



```
INSERT INTO table_name(column_list)
```

```
SELECT
```

```
    select_list
```

```
FROM
```

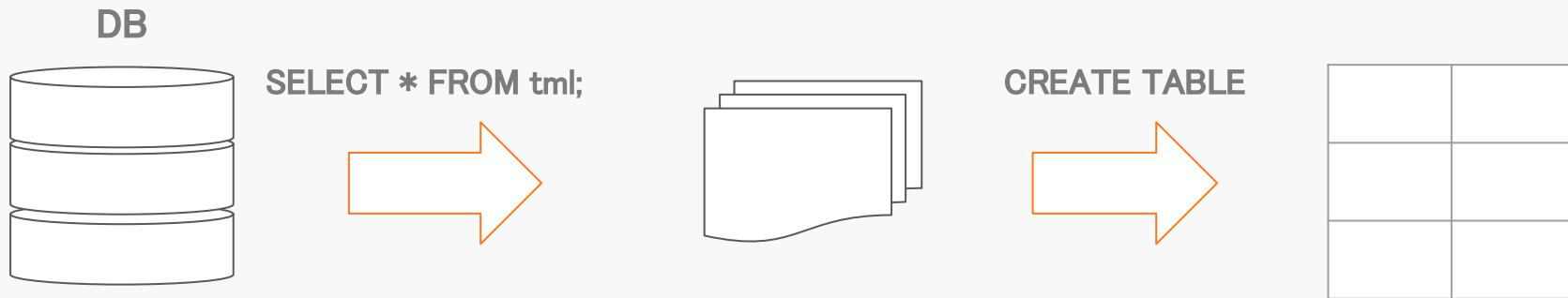
```
    another_table
```

```
WHERE
```

```
    condition;
```

# CREATE TABLE SELECTとは

別のテーブルを作成して、SELECT処理の実行結果を挿入する



```
CREATE TABLE table_name (AS)
SELECT
  select_list
FROM
  another_table
(WHERE
  condition);
```