はい、PARTITION BY と GROUP BY の違いについて詳しく解説します。

**PARTITION BY (ウィンドウ関数の一部):**

* **役割:** ウィンドウ関数 (OVER()句の中で使用) と共に使用され、データをグループ化して、グループごとの計算結果を各行に付与します。
* **動作:** 指定された列の値に基づいてデータをグループ化しますが、**行を集約しません。** つまり、元のテーブルの各行が結果セットに残り、各行にグループごとの計算結果が付加されます。
* **用途:** 各行に対して、グループ内の集計値、順位、割合などを表示したい場合に便利です。例えば、部署ごとの平均給与を従業員ごとに表示する場合や、売上ランキングを各行に表示する場合などに使用します。
* **構文:**
* SELECT
* 列名,
* ウィンドウ関数(引数) OVER (PARTITION BY 列名 ORDER BY 列名)
* FROM
* テーブル名

[with caution](https://support.google.com/legal/answer/13505487).SQL

* + PARTITION BY 列名: データをグループ化する基準となる列を指定します。
  + ORDER BY 列名: ウィンドウ関数内で順位を付ける際に、並び替えの基準となる列を指定します。

**GROUP BY (集計関数と共に使用):**

* **役割:** 集計関数 (SUM(), AVG(), COUNT(), MAX(), MIN(), etc.) と共に使用され、指定された列の値が同じ行をグループ化し、グループごとの集計結果を生成します。
* **動作:** 指定された列の値に基づいてデータをグループ化し、**グループごとに1行の結果を生成します。** つまり、元のテーブルの行数は結果セットで集約されます。
* **用途:** グループごとの集計結果（合計、平均、件数、最大値、最小値など）を取得したい場合に便利です。例えば、部署ごとの平均給与を計算する場合や、カテゴリーごとの売上合計を計算する場合などに使用します。
* **構文:**
* SELECT
* 列名,
* 集計関数(列名)
* FROM
* テーブル名
* GROUP BY
* 列名

[with caution](https://support.google.com/legal/answer/13505487).SQL

* + GROUP BY 列名: データをグループ化する基準となる列を指定します。

**主な違いのまとめ:**

|  |  |  |
| --- | --- | --- |
| 特徴 | PARTITION BY (ウィンドウ関数) | GROUP BY (集計関数) |
| **主な目的** | 各行にグループごとの集計結果などを付与 | グループごとの集計結果を算出 |
| **行の集約** | 行を集約しない (元のテーブルの行数が残る) | 行を集約する (グループごとに1行の結果) |
| **使用する関数** | ウィンドウ関数 (例: AVG() OVER(), RANK() OVER()) | 集計関数 (例: SUM(), AVG(), COUNT()) |
| **結果セット** | 元のテーブルの各行に対応した行を含む | グループごとに集約された行のみを含む |
| **代表的な例** | 部署ごとの平均給与を各従業員に表示 | 部署ごとの平均給与を表示 |

**具体例:**

**テーブル：従業員**

|  |  |  |  |
| --- | --- | --- | --- |
| 従業員ID | 従業員名 | 部署ID | 給与 |
| 1 | 田中 | 1 | 5000 |
| 2 | 佐藤 | 1 | 6000 |
| 3 | 鈴木 | 2 | 7000 |
| 4 | 高橋 | 2 | 8000 |
| 5 | 伊藤 | 1 | 5500 |

**例1: PARTITION BY (ウィンドウ関数)**

SELECT

従業員名,

部署ID,

AVG(給与) OVER (PARTITION BY 部署ID) AS 部署平均給与

FROM

従業員;

[with caution](https://support.google.com/legal/answer/13505487).SQL

**結果:**

|  |  |  |
| --- | --- | --- |
| 従業員名 | 部署ID | 部署平均給与 |
| 田中 | 1 | 5500 |
| 佐藤 | 1 | 5500 |
| 伊藤 | 1 | 5500 |
| 鈴木 | 2 | 7500 |
| 高橋 | 2 | 7500 |

各従業員に、その所属部署の平均給与が付与されています。元の行数は変化していません。

**例2: GROUP BY (集計関数)**

SELECT

部署ID,

AVG(給与) AS 部署平均給与

FROM

従業員

GROUP BY

部署ID;

[with caution](https://support.google.com/legal/answer/13505487).SQL

**結果:**

|  |  |
| --- | --- |
| 部署ID | 部署平均給与 |
| 1 | 5500 |
| 2 | 7500 |

各部署の平均給与が計算され、結果は部署ごとに1行に集約されています。

**まとめ:**

* PARTITION BY は、ウィンドウ関数と一緒に使い、各行に対してグループごとの集計値などを付与します。
* GROUP BY は、集計関数と一緒に使い、グループごとの集計結果を集約して取得します。
* どちらを使うかは、どのような結果セットを生成したいかによります。

PARTITION BY と GROUP BY の違いを理解することで、SQLクエリの表現力が向上し、複雑な分析処理も効率的に記述できるようになります。