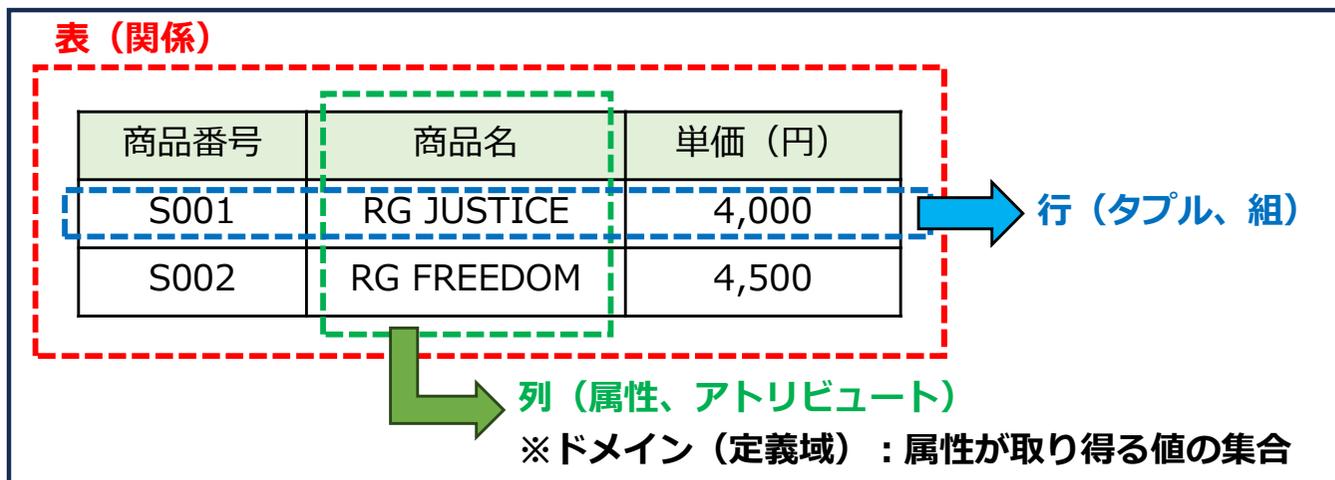


【関係演算を極める】

関係データベースの構図

今更ですが、関係演算を説明する前に、関係データベース（RDB）の構造を以下に示します。



関係演算 : 関係に対して行う8つの演算

まず、下表（関係演算その1）のとおり、一般的な集合演算で行われる和、差、積（共通）及び直積の4つがあります。そして、直積以外の和、差及び積は、和両立の条件を満たす必要があります。

和両立とは、関係Rと関係Sにおいて、属性の数（次数）が同じで、各属性のドメイン（定義域）が等しいものをいい、タプルの制約はありません。また、**SELECT文により和両立の条件を満たすようにしてから、和、差及び積を実行してもOKです。**

表 : 関係演算その1

関係演算	SQL	記法	特記事項
和	UNION	U、+	2つのSELECT文をUNION（重複行をまとめる）で繋ぐ。UNION ALLは重複行をそのまま表示するので、関係演算の「和」ではない。
差	EXCEPT	-	2つのSELECT文をEXCEPTで繋ぐ。EXCEPT ALLは上記に同じ。
積（共通）	INTERSECT	∩	2つのSELECT文をINTERSECTで繋ぐ
直積	CROSS JOIN	×	実際はFROM句で2つの関係の直積を実施

残りの4つの演算は、下表（関係演算その2）のとおり、関係代数特有の射影、選択、結合（下図参照）、商があります。

表（関係演算その2）

関係演算	SQL	記法	特記事項
射影	SELECT (DISTINCT)	なし	SELECT句に属性指定。DISTINCTによる重複の排除
選択	WHERE	なし	WHERE句に条件を指定
結合	別途説明（ 下図参照 ）		
商	なし	÷	商を直接求めるSQLはない。間接的はすごく長くなる。

結合演算

内部結合（ θ 結合）

これまでよく見てきたSELECT文の多くは内部結合です。つまり、「**SELECT * FROM R, S WHERE R.A>S.A**」において、FROM句で2つの関係の直積を求め、WHERE句で選択し、SELECT句で射影していたのです。なお、 θ とは比較演算子であり、「=、<>、<、>、 \leq 、 \geq 」等です。

また、上記のSELECT文は「**SELECT * FROM R INNER JOIN S ON R.A>S.A**」と同義です。

等結合

θ 結合において、比較演算子が「=」の場合の特殊形です。ただし、SELECT句で「*」とすると、**等結合した2つの属性（共通属性）の両方が表示されてしまう**ため、SELECT句で属性を指定することで回避します。

自然結合

等結合の特殊形です。共通属性の名前が完全一致ならば、「**SELECT * FROM R NATURAL JOIN S**」だけで、つまりONやUSINGを使用することなし（そもそも使用できない決まり）で、**共通属性の一方を削除して表示**してくれます。

外部結合

左外部（LEFT OUTER JOIN）、右外部（RIGHT OUTER JOIN）及び完全外部（FULL OUTER JOIN）があります。これらは「**SELECT * FROM R JOIN句 S ON R.A=S.A**」や「**SELECT * FROM R JOIN句 S USING A**」と書きます。

※ **INNER JOIN, LEFT OUTER JOIN, RIGHT OUTER JOIN, FULL OUTER JOIN**から**INNER**及び**OUTER**は省略可能