

データベース

第1正規形 (1NF)

ルール

- 繰り返し項目を持たない
- 全ての属性 (列) は原子的 (分割できない単位) であること

第2正規形 (2NF)

ルール

- 第1正規形を満たすこと
- 主キーが複合キーの場合、主キーの一部にのみ依存する部分関数従属を排除する

第3正規形 (3NF)

ルール

- 第2正規形を満たすこと
- 推移的関数従属を排除する (キー以外の属性が、他の非キー属性に依存してはいけない)

SoC(System on a chip) CPU,GPU,DSP,メモリ,タイマ,通信モジュールなどを一つのチップ上に集約したシステムLSI(大規模集積回路)。

◆ ACID 特性とは

データベースにおける **トランザクションの性質** を表す4つの頭文字です。

1. Atomicity (原子性)

- トランザクションの処理は **全て実行されるか、全く実行されないか** のどちらか。
- 例：銀行振込で「送金元から引き落とし」だけ行われて「送金先に入金」されない、という中途半端な状態は起こらない。

1. Consistency (一貫性)

- トランザクション実行前後で、データの整合性ルール (制約、ビジネスルール) が保たれる。
- 例：商品の在庫がマイナスになるような状態は許されない。

1. Isolation (独立性、分離性)

- 複数のトランザクションが同時に実行されても、**互いに干渉せず、直列に実行したのと同じ結果** になる。
- 例：二人が同時に同じ商品を購入しても、在庫数が正しく管理される。

1. Durability (永続性)

- コミットしたトランザクションの結果は、障害が発生しても失われない。
- 例：電源が落ちて、直前までの取引結果がログやディスクに残っている。

◆ ビューが更新可能になる条件

1. 元が単一テーブル

- ビューは1つのテーブルから作られていること

- 複数テーブルの JOIN を含むと基本的に更新不可
- 2. **主キーを保持している**
 - 一意に行を識別できる列 (主キー) がビューに含まれていること
- 3. **集計や演算を含まない**
 - SUM、AVG、COUNT などの集計関数を含むビューは更新不可
 - DISTINCT や GROUP BY も不可
- 4. **行が一意に対応する**
 - ビューの1行が元テーブルの1行に直接対応している必要がある
 - 例: WHERE で絞り込んでいる場合は OK
- 5. **サブクエリや UNION を含まない**
 - 複雑な SQL で作られたビューは更新不可

◆ ビュー (View) とは

- 既存のテーブルから作られる仮想的なテーブル
 - 実データを持たず、SQL (SELECT 文) の結果を「名前付きテーブル」として扱う
 - 利用者から見ると通常のテーブルのように操作できる
- 「実体は SELECT 文、見た目はテーブル」。

CAP 定理 -> 分散処理システムでは、一貫性 (Consistency) ・可用性 (Availability) ・分断耐性 (Partition Tolerance) の3つの特性のうち、最大でも同時に二つまでしか満たせない

SQL の単語

% は SQL では任意の 0 文字以上

DISTINCT

SQL の **DISTINCT** は、重複する行を取り除いて、ユニークな行だけを返す ためのキーワード

RESTRICT

参照する側の行が残っている場合には、参照される行の更新、削除をできないようにする

CASCADE

参照される側の行が削除・更新されたら、参照している行も削除、更新

SET DEFAULT

参照する行側にデフォルト値をつける

UNION SELECT 文の結果を和演算 (重複なし) にする

UNION ALL は重複あり

INTERSECT 共通部分

CROSS JOIN 直積

◆ 主キー (Primary Key)

- 複数ある候補キーの中から **1つを選んだもの**
 - データベース設計上「このキーで行を識別する」と決めたキー
 - 主キーには必ず **NOT NULL制約** と **一意性制約** が課される
- 上の例なら、管理上は「社員番号」を主キーに選ぶのが一般的。

◆ 候補キーと主キーの関係

- 候補キー = キーになれる可能性があるもの（複数あり得る）
- 主キー = その中から **1つ選ばれたもの**

◆ その他関連用語

- 外部キー (Foreign Key)
 - 他のテーブルの主キーを参照するキー
 - 例：「注文テーブル」の 社員番号 が「社員テーブル」の主キーを参照する
- 代替キー (Alternate Key)
 - 候補キーのうち、主キーに選ばれなかったもの
 - 上の例なら、主キーを「社員番号」とした場合、メールアドレス が代替キー

◆ まとめ表

用語	意味	数
候補キー	一意に識別できる属性の組み合わせ	複数あり得る
主キー	候補キーの中から選んだ1つ	必ず1つ
代替キー	主キー以外の候補キー	0個以上
外部キー	他テーブルの主キーを参照するキー	複数あり得る

◆ 違いのまとめ

項目	DISTINCT	UNIQUE
種類	SELECT文のキーワード	制約（テーブル定義）
効果	結果セットの重複を排除	データそのものを重複禁止にする
影響範囲	出力だけ（DBには影響なし）	テーブルの格納データに影響
用途	クエリ結果をユニークにしたい	列の値をユニークに保ちたい

◆ 2相コミットの流れ

登場人物：

- コーディネータ（調整役）

- 参加者 (各データベースやサーバ)

フェーズ1：準備 (投票要求フェーズ, Prepare Phase)

1. コーディネータが各参加者に「コミット準備できる？」と問い合わせる (prepare)。
2. 参加者はローカルでトランザクションを実行して、結果をログに保存し「コミット可能」または「ロールバック必要」と返答。

フェーズ2：コミット (コミットフェーズ, Commit Phase)

- 全員が「Yes (コミット可能)」なら → コーディネータが「コミット実行」命令を送る → 参加者は確定処理。
- 1人でも「No (失敗)」なら → コーディネータが「ロールバック」命令を送る → 全員が取り消す。

◆ 例 (銀行振込)

顧客Aが銀行Xから銀行Yへ送金する場合：

- 銀行X (残高減算)、銀行Y (残高加算) が同時に成功する必要がある。
- もしXだけ成功してYが失敗したら整合性が壊れる。
→ 2PCで「両方成功なら確定、片方失敗なら両方取り消し」とする。