

# 【正規形を振り返って】

## 「主キー」と「候補キー」

振り返ってみると、正規化において、主役は「主キー」でなく「候補キー」でした。その理由は、正規化の過程において「主キー」になり得るもの、つまり、複数の「候補キー」全てが検討の対象となるためです。よって「候補キー」を理解していないと正規化はできません。また「主キー」には「非NULL制約」がありますが「候補キー」には「非NULL制約」がないため、正規化において、使いやすい面もあります。

そこで、下図（主キー等のまとめ）のとおり、今一度「NULL」「主キー」「候補キー」について、まとめてみました。

### NULL（ヌル or ナル）

- NULLは、英語で、不明や未知を表す**Unknown**である。例として「メアドはあるらしいが、それを知らない」
- NULLは、英語で、非存在や適用不能を表す**Inapplicable**である。例として「メアドがない」
- NULLは値ではない。そのため、比較はできず、SQL文では「列名=NULL」ではなく「列名 IS NULL」となる。
- 属性のデータ数をカウントする時、NULLの行はカウントしない。

### 主キー

候補キーの中から1つを選んだものである。以下の①及び②の制約がある。

- ①一意制約（他の行と重複しない）
- ②非NULL制約（主キーの真部分集合であってもNULLは不可）

### 候補キー

- 主キーの候補であり、1つ以上存在する。
- スーパーキーの内、行を一意に特定できる必要最小限の複数又は単独の属性で構成される。行を一意に特定できればいいので、以下の①又は②のとおり、非NULL制約はない。
- ①単独：NULLは1つ、NULL以外は行を一意に特定できる。だから、1つあるNULLも、行を一意に特定できる。
- ②複数：1つの属性に複数のNULLが存在しても、残りの属性によって、行を一意に特定できる。
- 主キーの候補を候補キーと呼ぶのならば、候補キーも非NULL制約を課すべきという議論もあるが、複数の候補キーから主キーを選定する基準と考える程度でいいかもしれない。

## 図：主キー等のまとめ

## 自明な関数従属性

ボイス・コード正規形のところで「**X → Yは自明な関数従属性である**」を詳しく説明しましたが、振り返ってみると、非正規形から第1正規形にする際に、既に体験していたことに気付きました。つまり「**繰り返し項目**」を「**単一値**」にする時です。

先の投稿では表の情報量が多かったので、下図（自明な関数従属性の振り返り）のとおり、単純な表により説明します。

「**単一値でない**」は「**繰り返し項目がある**」と同じ意味であり、以下のとおり、属性 {社員ID、身長、体重} に係る1行に対して、属性 {趣味} が繰り返し項目になっている。この繰り返し項目が存在する状態は「**非原子定義域（非原子ドメイン）**」や「**非単純定義域（非単純ドメイン）**」と呼ばれる。

言い換えると、主キー {社員} に対して {趣味} は**一意に決まらない**、つまり、**関数従属性がない**ことになる。

社員ID	身長	体重	趣味
A001	175	65	プラモデル サッカー

 又は 

社員ID	身長	体重	趣味
A001	175	65	{プラモデル、サッカー}

よって、以下のとおり {趣味} を基準に表を再構築することで {趣味} は「**繰り返し項目でない**」状態、つまり「**単一値**」「**原子定義域**」「**単純定義域**」となる。

この時、主キー {社員ID、趣味} とすることで {社員ID、趣味} → {趣味} となり「**自明な関数従属性**」となる。

社員ID	身長	体重	趣味
A001	175	65	プラモデル
A001	175	65	サッカー

図：自明な関数従属性の振り返り