

ダウンロード版

データベース問題 (3)

学籍番号

氏名

学年

学籍番号、氏名は全員記入をお願いします。

1. 第一正規形の定義を書け
2. 第一正規形で無いリレーションの例を2次元の表の形で書きなさい。
3. キー(key) (あるいは候補キー(candidate key) ともいう) の定義を書け
※ この種類の問題では、解答を省略しない方がよい (重要部分が抜けていると、本試験では0点になる)
4. 大学における学生のデータベースを考える。リレーション名「学生」を作り、「氏名」という属性名を作る (リレーションスキーマは、「学生 (氏名)」になる)。この場合「氏名」は「学生」の主キーにはなりえない。その理由を説明せよ。

5. 大学における学生のデータベースを考える．今度は，「氏名」，「住所」，「入学年度」という 3 つの属性名を作る（リレーションスキーマは，「学生（氏名，住所，入学年度）」になる）．このリレーションの主キーは何が良いか考えよ．その理由を説明せよ．

※ 同じ住所に，同じ氏名を持つ異なる学生（同姓同名の学生）は，現実にはいないと考えても妥当だろう．

6. キーと主キーの違いは何か？

7. 大学における学生のデータベースを考える．今度は，「学籍番号」，「氏名」，「住所」，「入学年度」という 4 つの属性名を作る（リレーションスキーマは，「学生（学籍番号，氏名，住所，入学年度）」になる）．このリレーションの主キーは何が良いか考えよ．その理由を説明せよ．

8. 外部キーの定義を書け

※ この種類の「知識を問う問題」では，解答を省略せず，知っていることをきちんとした文章で記述すべきである（重要部分が抜けていると，本試験では 0 点になるため）

1. 空値(NULL)は、どのような場合に用いられるかを例を使って説明せよ
8. データベースにおける一貫性制約は何の役に立つか？

<必須の用語>

■ 単純値 (単純のことを「シンプル」ともいう)

分解不可能な値のこと。普通は「数値や文字や文字列や時刻や日付や論理値などの値」という意味で使う。直積集合の要素や、べき集合の要素は単純値ではないと考えるのが普通である。

直積集合の例： 属性名「family name」のドメインと、属性名「given name」のドメインの直積集合

べき集合の例： ドメイン {みかん, りんご, バナナ} から構成されるべき集合は, { ϕ , {みかん}, {りんご}, {バナナ}, {みかん, りんご}, {りんご, バナナ}, {バナナ, みかん}, {みかん, りんご, バナナ}}

¥

■ 第一正規形

ドメインとして、分解不可能な「単純値」のみを対象とすること。

■ キー (「候補キー」ともいう)

リレーショナルデータモデルで「キー」というときは、次の意味である。

リレーションのタプルを唯一に識別するのに使える属性、あるいは、属性の組のうち極小なものをキーという。

キーである属性、あるいは、属性の組については、リレーションの 2 つ以上のタプルが、いかなる場合であっても、同一の属性値を持つことはありえない。

■ 主キー

リレーショナルデータモデルにおいて、ある 1 個のリレーションスキーマに対して、キーは複数種類ありえる。その中で、データベース管理者が、データベースの管理上最も適当と判断し、かつ、空値をとることがありえないものを、リレーションスキーマの設計時に選んだものが主キーである (つまりキーと主キーは違う)。あるリレーションスキーマに対して「キー」が 1 種類しかないときは、それが主キーになる。

※ 主キーとして選ばれたキーの属性、あるいは、属性の組は空値ではありえないことに注意。

■ 空値 (「NULL」とも呼ぶ)

空値はさまざまな意味で使われる。

・本来、値が存在しない・・・例えば、自由席の場合には「予約座席番号」が存在しない。「送付を希望しない」

ときは住所が空欄。

・未定、空席・・・例えば、「データベース」の授業の担当者がまだ決まっていないようなとき

・値が未知・・・例えば、住所がまだ分からないとき (住所は存在するが、まだ知らない)

■ 外部キー

リレーションスキーマ $R(..., A_i, ...), S(..., B_j, ...)$ について、属性 B_j がリレーション S の主キーであるとする。 R と S のインスタンス r および s について、以下の条件が成り立つとき、属性 A_i をリレーション R の外部キーであるという。

r 中の任意のタプルが持つ A_i の値が、

(1) S のインスタンス s 中のタプルが持つ B_j の値になっている。あるいは

(2) 空値である。

<定義はここまで>

※ 上記の定義では、キーや主キーになっている属性の個数が 1 個だと仮定しているが、実際には、キーや主キーは複数の属性の組であってよい。

■ 一貫性制約

データベースが実世界を正しく反映しているときに満足せねばならない制約条件のこと。例えば、誕生年として「3009」が記録されているようなデータベースは、とても実世界を正しく反映しているとは言えず、信用出来ない。誕生年には、例えば、「0 以上 2009 以下」のような制約条件があるはず（値の範囲は、データベースの用途によっても変わってくることに注意。会員名簿なら「1889 以上 2009 以下」のようになるだろう。今後 10 年間データベースを使い続けると分かっているなら「1889 以上 2019 以下」のようになるだろう）。

リレーショナルデータモデルでの一貫性制約とは、リレーションスキーマのインスタンスの任意のタプル（言い換えると、リレーションスキーマのインスタンスの各要素）の各成分の値が、満足せねばならない制約条件のこと。

リレーショナルデータベース管理システムには、普通、一貫性制約を記述できる機能、記述された一貫性制約に反するような更新を受け付けなくする機能がある。

リレーショナルデータモデルでの一貫性制約は種々の種類がある。例示すると下記の通り。

・キー制約

キーであると指定された属性（あるいは属性の組）や、主キーであると指定された属性（あるいは属性の組）については、キーあるいは主キーとしての条件を満足せねばならないという制約

・外部キー制約

外部キーであると指定された属性（あるいは属性の組）が、外部キーとしての条件を満足せねばならないという制約

・非空制約

非空であると指定された属性は、空値（NULL）になってはいけないという制約

・ドメイン制約

リレーションのタプルの各成分の値は、対応する属性のドメインの要素でなければならないという制約。

（例）「試験の点数」という属性のドメインが「{x|x は 0 以上 100 以下の整数}」であると定められているとき、試験の点数の値が、この範囲外になってはいけない。

※ 「ドメイン」とは属性が取りえる値の集合のこと（第 2 回授業の内容）。

・従属性に関する制約

※ 「従属性」について、授業で未だ説明していないので、従属性に関する制約の説明は省略する。