

リレーショナルデータベースモデル

— 構造記述 —

リレーショナルデータモデル

- 1970年 E.F.Codd が提案
- データベースをリレーション (関係) の集まりとしてモデル化したもの
- 数学的概念に基づいている
- 商用だけでなくフリーのリレーショナルデータベースのDBMSがある
 - Oracle Database, Microsoft SQL Server, ...
 - MySQL, PostgreSQL, ...

リレーションとは

ドメインの直積集合の有限部分集合

- ドメイン (domain, 定義域):

- 何らかの集合 (名前, 身長, 電話番号, ...)

D_1, D_2, \dots, D_n で表現する 例: $D_1 = \text{CHAR}(10), D_2 = \text{INTEGER}, \dots$

- 直積

(direct product あるいは Cartesian product)

- 要素の全ての組み合わせからなる集合

例えば, { 家康, 秀忠, 家光 } と { 初代, 二代, 三代 } の直積は,

{ (家康, 初代), (家康, 二代), (家康, 三代), (秀忠, 初代), (秀忠, 二代), (秀忠, 三代),
(家光, 初代), (家光, 二代), (家光, 三代) }

- n 個のドメインの直積は $D_1 \times D_2 \times \dots \times D_n$ と記述される

• リレーション

➤ 繰り返しになるが,

ドメイン D_1, D_2, \dots, D_n 上のリレーション R とは
直積 $D_1 \times D_2 \times \dots \times D_n$ の任意の有限部分集合

直積は無限集合になることがあるが, リレーションは有限集合である

➤ R の各要素を **タプル** (tuple) という

➤ タプルの総数を R の濃度あるいは基数 (cardinality) という

➤ ドメインの **個数** を **次数** (degree) という

✓ 次数1のリレーションを **単項リレーション**, 次数 n ($n \geq 2$) のリレーションを **n 項リレーション** という

科目を表すドメインとして, 科目名 (4個), 科目番号 (999個), 単位数 (3個), 担当者 (10名) があるとするなら,

科目名 = { 中級プログラミング, データ構造とアルゴリズム, アルゴリズム演習, データベース }

科目番号 = { 5005001, 5005002, ..., 5005999 }

単位数 = { 1, 2, 8 }

担当者 = { 安藤, 香川, 亀井, 喜田, 米谷, 最所, 高木, 富永, 福森, 八重樫 }

直積集合の要素数は?

$$4 \times 999 \times 3 \times 10 = 119,880$$

ドメインが整数や実数なら?

無限

右のリレーション (授業科目) は,
直積集合の部分集合

授業科目			
科目名	科目番号	単位数	担当者
中級プログラミング	5005030	2	富永
中級プログラミング	5005030	2	安藤
中級プログラミング	5005030	2	香川
データ構造とアルゴリズム	5005070	2	安藤
アルゴリズム演習	5005080	1	安藤
アルゴリズム演習	5005080	1	最所
データベース	5005160	2	最所
⋮	⋮	⋮	⋮

リレーションスキーマ

- リレーションが持つ**普遍的な構造や意味** (ある時点だけではなく、**時間的に不変**) を抽出したもの
- $R(A_1, \dots, A_n)$ で表す
 - R : **リレーション名**
 - A_1, \dots, A_n : **属性名**
 - ✓ 属性名は一意
 - ✓ n は属性の個数で、リレーションスキーマの**次数**
 - ✓ 属性 A_i のとり得る値の集合を**ドメイン** (あるいは**定義域**) と呼び、ここでは D_i と記述する

ドメイン関数 (属性のドメインを表す関数)

$$\text{dom} : A_i \rightarrow D_i (i = 1, 2, \dots, n)$$

インスタンス

- ある時点でのリレーションそのもの
 - 実際に格納されているデータ
 - 時間的に変化する
- リレーションスキーマとインスタンス

授業科目			
科目名	科目番号	単位数	担当者
中級プログラミング	5005030	2	富永
中級プログラミング	5005030	2	安藤
中級プログラミング	5005030	2	香川
データ構造とアルゴリズム	5005070	2	安藤
アルゴリズム演習	5005080	1	安藤
アルゴリズム演習	5005080	1	最所
データベース	5005160	2	最所
⋮	⋮	⋮	⋮

表によるリレーションの表現

- 行 (row)

- **タプルに対応** (順序に意味はない)
- 各ドメインに属する値 (属性値) の組

- カラム (column)

- 表の縦の列
- **属性に対応** (属性で指定する場合は順序に意味はない)

- リレーションは集合である

- **同じ値を持つタプルが1つのリレーション中に存在してはならない**



1つの表に**属性値の組が同じ行は存在しない**

授業科目			
科目番号	科目名	単位数	担当者
5005030	中級プログラミング	2	富永
5005030	中級プログラミング	2	安藤
5005030	中級プログラミング	2	香川
5005070	データ構造とアルゴリズム	2	安藤
5005080	アルゴリズム演習	1	安藤
5005080	アルゴリズム演習	1	最所
5005160	データベース	2	最所
⋮	⋮	⋮	⋮

第1正規形 (the first normal form, 1NF)

- リレーショナルデータモデルでは、ドメインとして分解不可能な単純値のみの集合を対象とする
- 1つのタプルの属性に複数の属性値を含まない
- 第1正規形制約を解除したリレーションを非第1正規形リレーションと呼ぶ

科目

科目番号	科目名	単位数	担当者
5005030	中級プログラミング	2	富永
5005030	中級プログラミング	2	安藤
5005030	中級プログラミング	2	香川
5005070	データ構造とアルゴリズム	2	安藤
5005080	アルゴリズム演習	1	安藤
5005080	アルゴリズム演習	1	最所
5005160	データベース	2	最所
⋮	⋮	⋮	⋮

第1正規形

科目

科目番号	科目名	単位数	担当者
5005030	中級プログラミング	2	富永
			安藤
			香川
5005070	データ構造とアルゴリズム	2	安藤
5005080	アルゴリズム演習	1	安藤
			最所
5005160	データベース	2	最所
⋮	⋮	⋮	⋮

非第1正規形