明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

中村研究室ゼミ
データベース

中村 聡史

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

大規模データの管理・運用

- 明治大学の全学生，全教職員の情報（名前，住所，連絡先など）を管理するシステム
- 膨大な患者のカルテを管理するシステム
- 遺伝子データを管理するシステム
- 判例を管理するシステム
- 論文を管理するシステム
- 天気のデータを管理するシステム
- 住所を管理するシステム

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

膨大な情報の管理

- どのようにして管理するか？
- 紙ベースで管理可能？
  - 明治大学の男子学生の数は何人？
  - 単位が揃っていない学生の数は何人？
  - 学籍番号が＊＊＊＊な人の名前は？
  - 東京都の人口は何人？
  - 東京都で2011年に発生した事件の数は？
  - 中野区でインフルエンザにかかって病院で治療した20歳以下の女性の数は何人？

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

自分で作る？

- プログラムを作るのは容易ではない
  - ある条件を満たす人の数を数える
  - あるIDの人の名前を調べる
  - 給与の合計を出す
  - 商品の受発注を在庫を確認しながら受け付ける
  - 銀行の貯金管理

- 大変だし時間の無駄

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

関係データベース

- 表形式のデータベース
  - 属性（アトリビュート）と組（タプル）
  - 主キー（候補キー）と外部キー

No	姓	名	性別	誕生日	所属	・・・
00001	中村	聡史	男	******	******	・・・
00002	浅野	泰仁	男	******	******	・・・
00003	稲葉	利江子	女	******	******	・・・
00004	木村	欣司	男	******	******	・・・
00005	矢作	日出樹	男	******	******	・・・
00006	山肩	洋子	女	******	******	・・・
：	：	：	：	：	：	

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

関係データベース

- 複数の表を組み合わせて結果とする

顧客NO	名前	年齢	購買した商品
001001	中村 聡史	33	地鶏もも肉
001001	中村 聡史	33	ブルーチーズ
001001	中村 聡史	33	フランス産赤ワイン

顧客NO	名前	年齢
001001	中村 聡史	33
001002	浅野 泰仁	34
001003	田中 克己	58

顧客NO	購買した商品
001001	地鶏もも肉
001001	ブルーチーズ
001001	フランス産赤ワイン
001002	烏龍茶
001002	惣菜弁当
001003	食パン

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

データベースでは…

- 膨大なデータのやり取りが行われる
  - 銀行の決済やAmazonや楽天などの受発注および在庫管理，ライブチケット販売受付などどうやって整合性を取りつつ管理するか？
    - 1つしか席がないのに2人から同時に予約が来たらどうするか？
    - 家庭の口座に残金が10万円しか無いが，二人が同時に8万円を引落そうとしたらどうなるか？

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

ACID

- ジム・グレイ（1970年代後半）
- 信頼性のあるトランザクションが持つべき性質

Atomicity: 原子性

Consistency: 一貫性

Isolation: 独立性

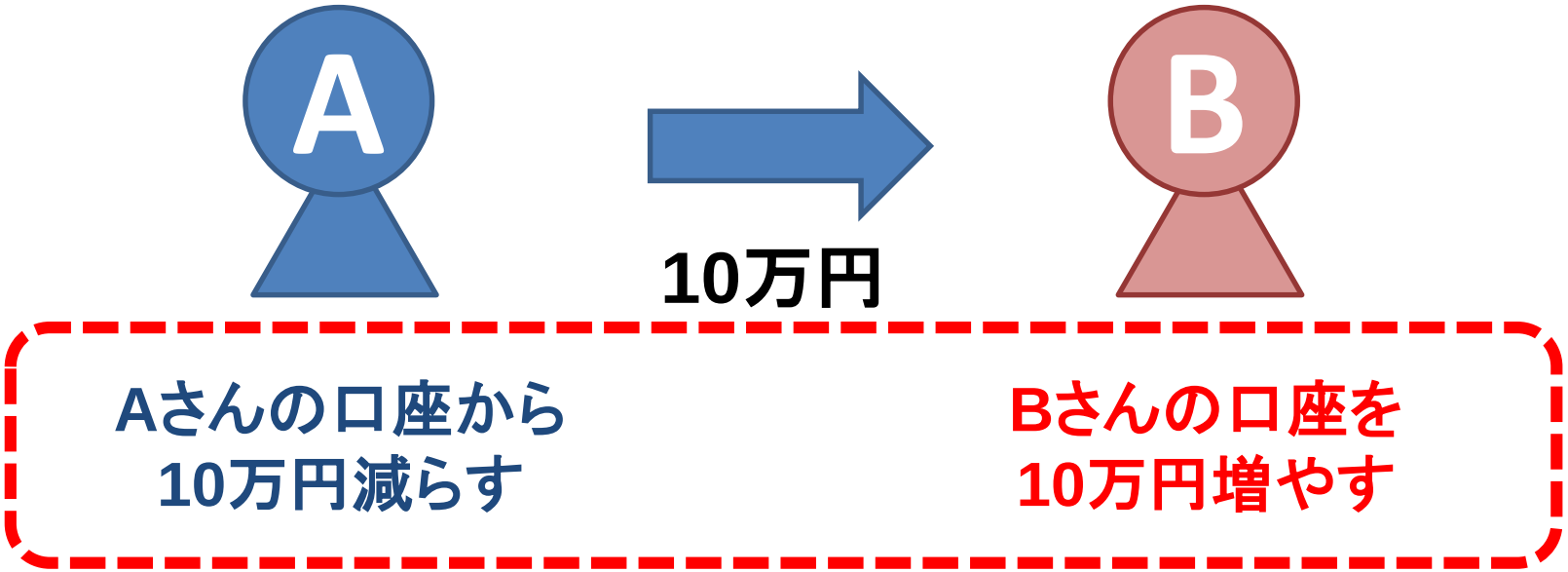
Durability: 永続性

http://ja.wikipedia.org/wiki/%E3%83%95%E3%82%A1%E3%82%A4%E3%83%AB:Jim_Gray_Computing_in_the_21st_Century_2006.jpg

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

トランザクション

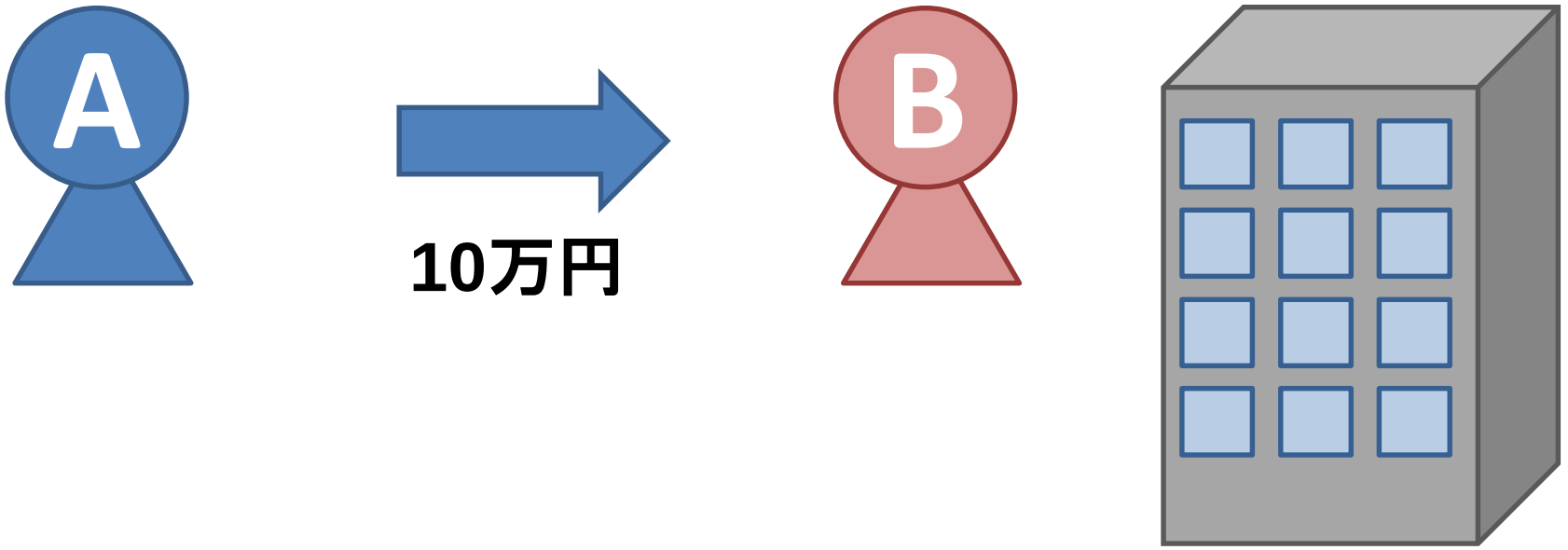
- データに対する1つの論理操作
- セットとなる処理
- タスクの集合



明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

Atomicity: 原子性

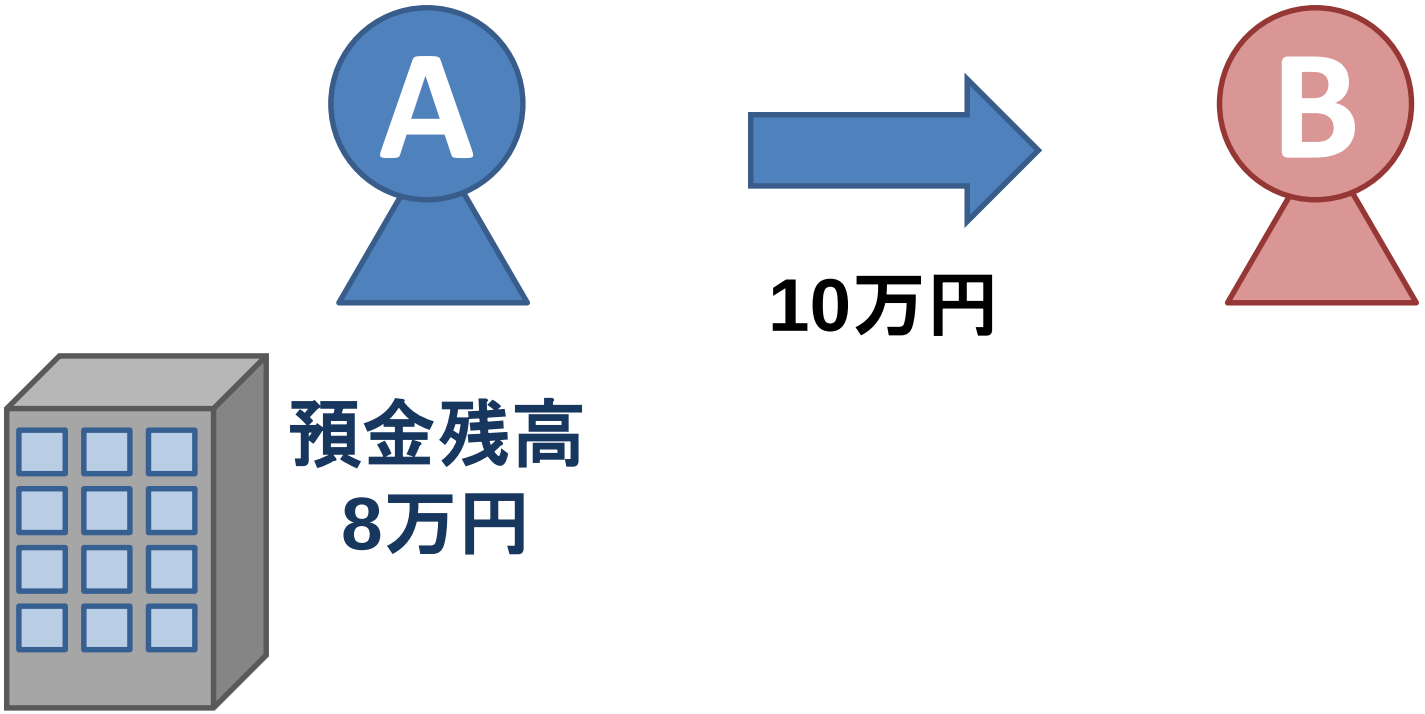
- トランザクション内のタスクを全て処理するか，全て処理しないかのどちらか



明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

Consistency: 一貫性

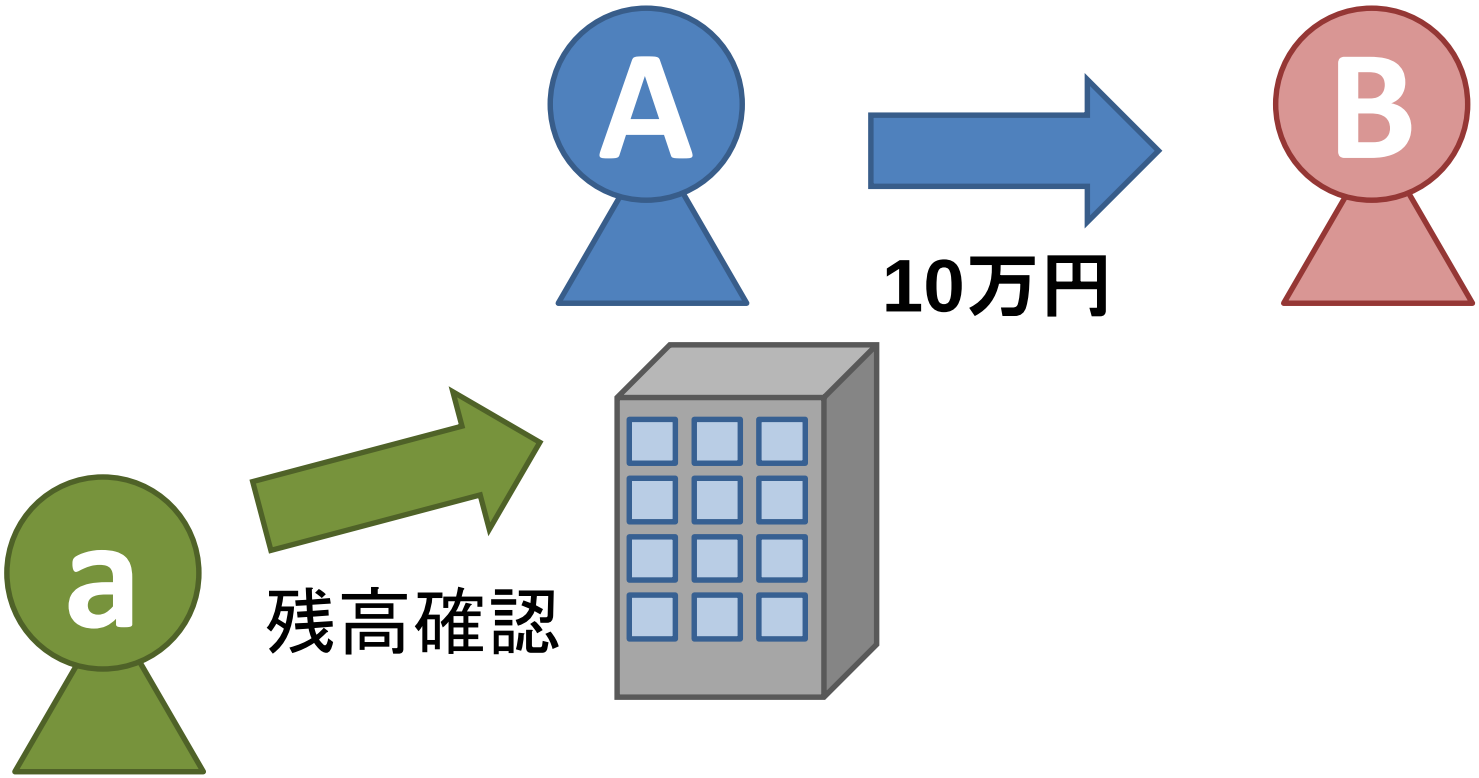
- トランザクションの前後で予め与えられた整合性を満たすようにする



明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

Isolation: 独立性

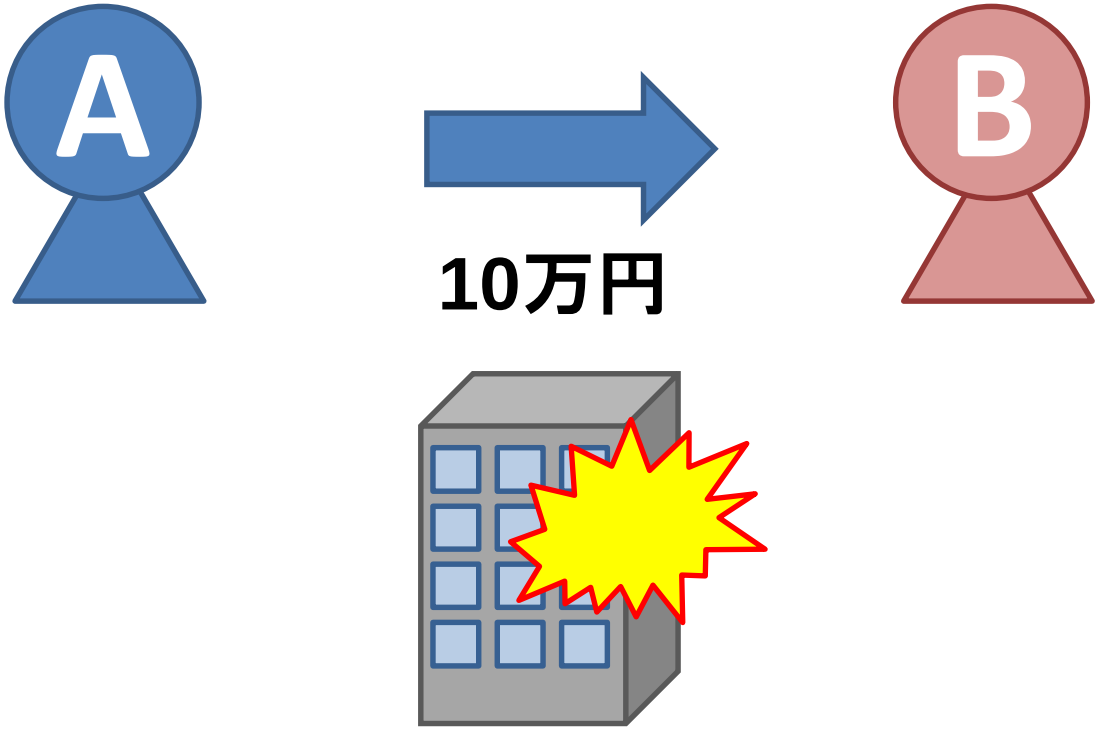
- トランザクション中の処理の過程は他から隠蔽される



明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

Durability: 永続性

- 完了したトランザクションは失われてはならない
- 異常時も完了分は復旧させる



明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

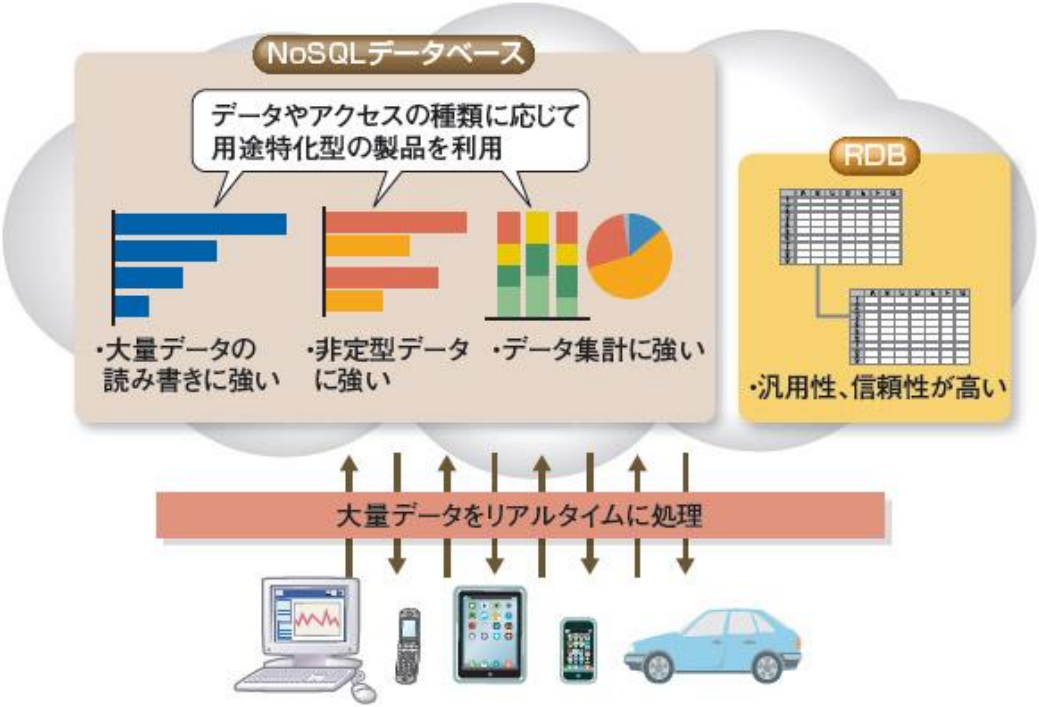
当然ですが・・・

- ACIDを完全に保証するトランザクション処理の実現は難しい
- ある程度緩めて実装

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

NoSQL

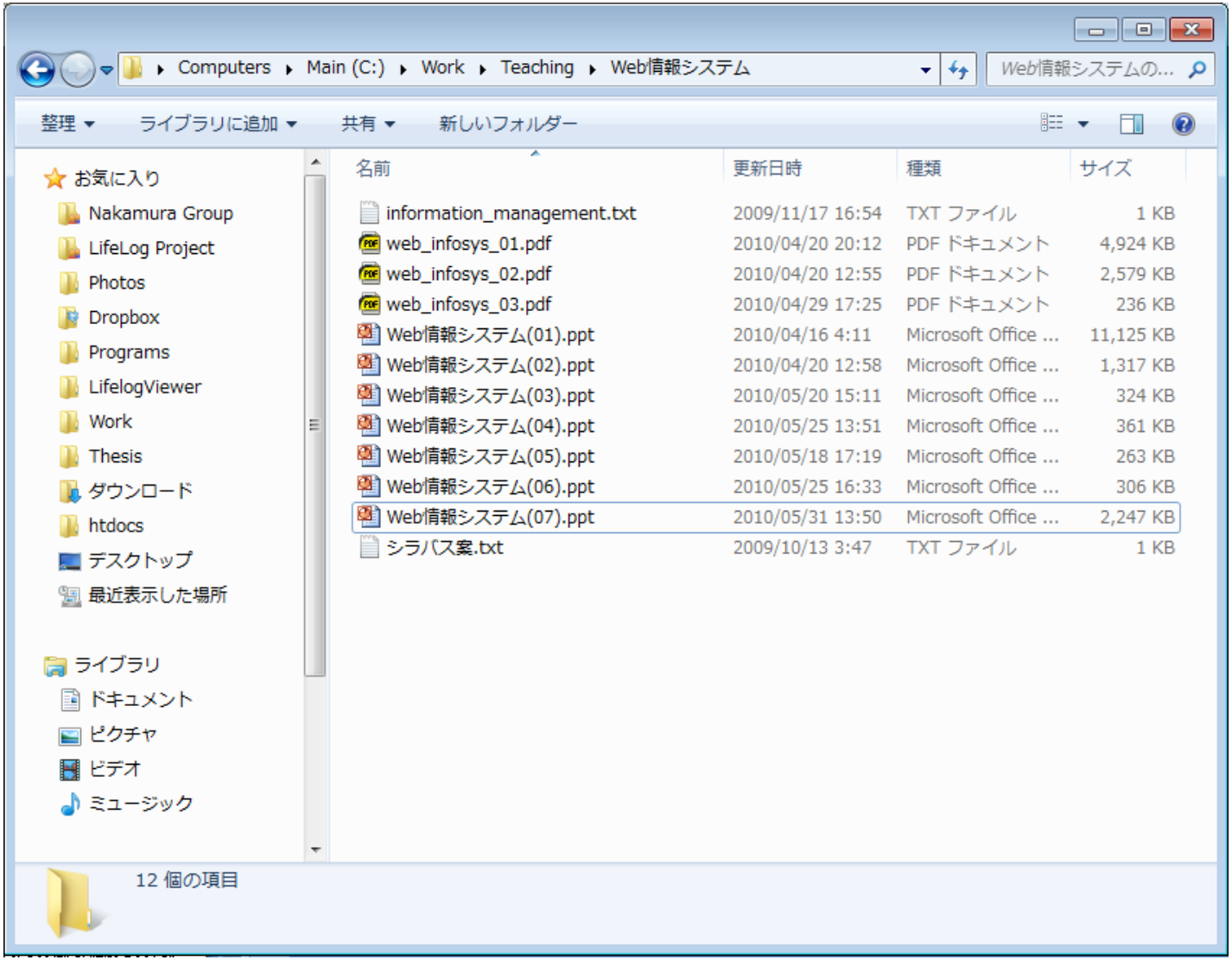
- SQLを否定するものではなく，Not only SQL
- RDBMSは並列化などスケールしにくい
- 並列化するための仕組み



 http://itpro.nikkeibp.co.jp/article/COLUMN/20110713/362387/?SS=imgview&FD=54139247&ST=system

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

身近なデータベース



明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

データベース管理システム

- Relational Database Management System
  - Oracle Database
  - Microsoft SQL Server
  - PostgreSQL
  - MySQL
  - SQLITE
  - DB2
  - FileMaker

  などなど

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

関係データベースに対する演算

- 制限（いくつかの組の抽出）
- 射影（いくつかの属性の抽出）
- 結合（ある属性でテーブルAとBをくっつける）
- 直積（テーブルAとBの組合せの全パターン）
- 和（重複を除くテーブルAとBの組すべて）
- 差（テーブルAからBに属する組を除いたもの）
- 交わり（テーブルAとBで一致する組を抽出）

演算結果は関係（関係テーブル）
演算を多段階に処理可能

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

制限 (selection)

顧客NO	購買した商品
001001	地鶏もも肉
001001	ブルーチーズ
001001	フランス産赤ワイン
001002	烏龍茶
001002	惣菜弁当
001003	食パン

顧客NO	購買した商品
001001	地鶏もも肉
001001	ブルーチーズ
001001	フランス産赤ワイン

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

射影 (projection)

顧客NO	名前	年齢	購買した商品
001001	中村 聡史	33	地鶏もも肉
001001	中村 聡史	33	ブルーチーズ
001001	中村 聡史	33	フランス産赤ワイン
001002	浅野 泰仁	34	烏龍茶
001002	浅野 泰仁	34	惣菜弁当
001003	田中 克己	58	食パン

顧客NO	名前	購買した商品
001001	中村 聡史	地鶏もも肉
001001	中村 聡史	ブルーチーズ
001001	中村 聡史	フランス産赤ワイン
001002	浅野 泰仁	烏龍茶
001002	浅野 泰仁	惣菜弁当
001003	田中 克己	食パン

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

結合 (join)

顧客NO	名前	年齢
001001	中村 聡史	33
001002	浅野 泰仁	34
001003	田中 克己	58

顧客NO	購買した商品
001001	地鶏もも肉
001001	ブルーチーズ
001001	フランス産赤ワイン
001002	烏龍茶
001002	惣菜弁当
001003	食パン

顧客NO	名前	年齢	購買した商品
001001	中村 聡史	33	地鶏もも肉
001001	中村 聡史	33	ブルーチーズ
001001	中村 聡史	33	フランス産赤ワイン
001002	浅野 泰仁	34	烏龍茶
001002	浅野 泰仁	34	惣菜弁当
001003	田中 克己	58	食パン

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

演習

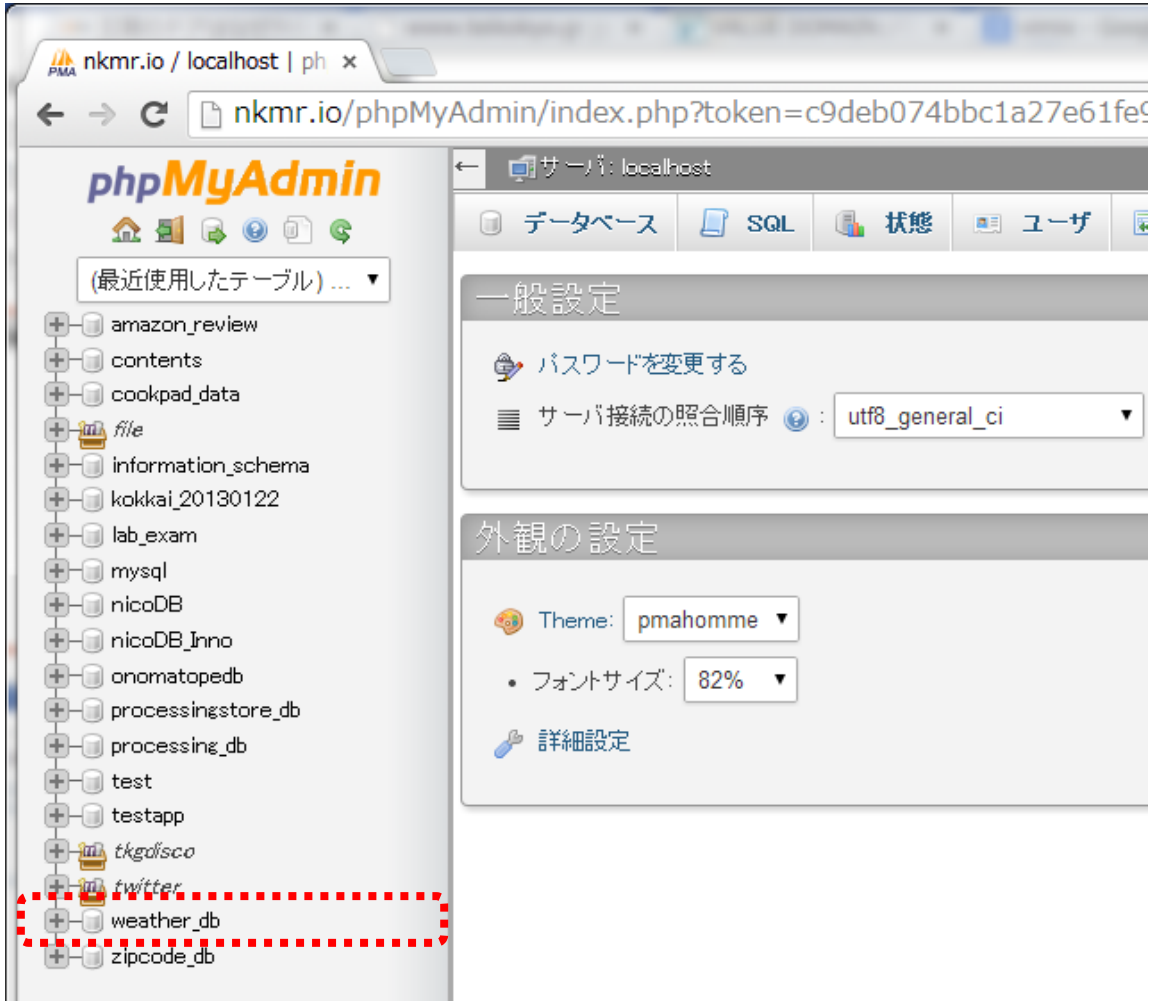
- xamppをインストールして http://localhost/phpMyAdmin/ にアクセス

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

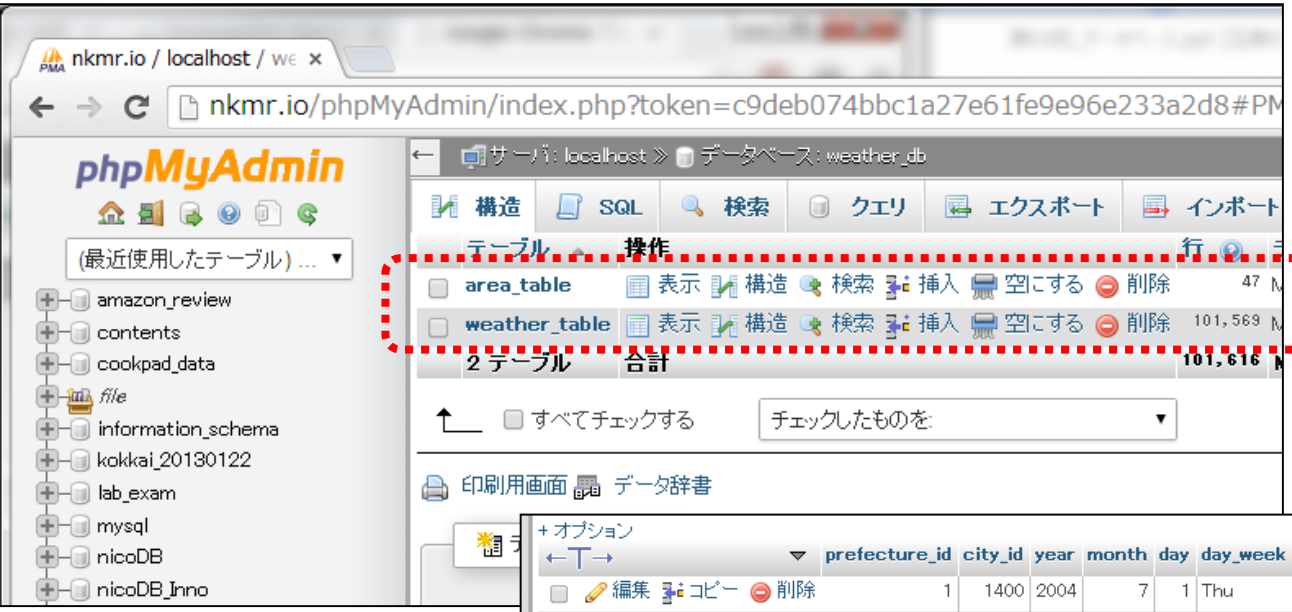
まずは

- weather_db というデータベースを選択しよう！



明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

weather_table/area_table



明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

weather_table

Yahoo!天気から情報を集め，変換しています

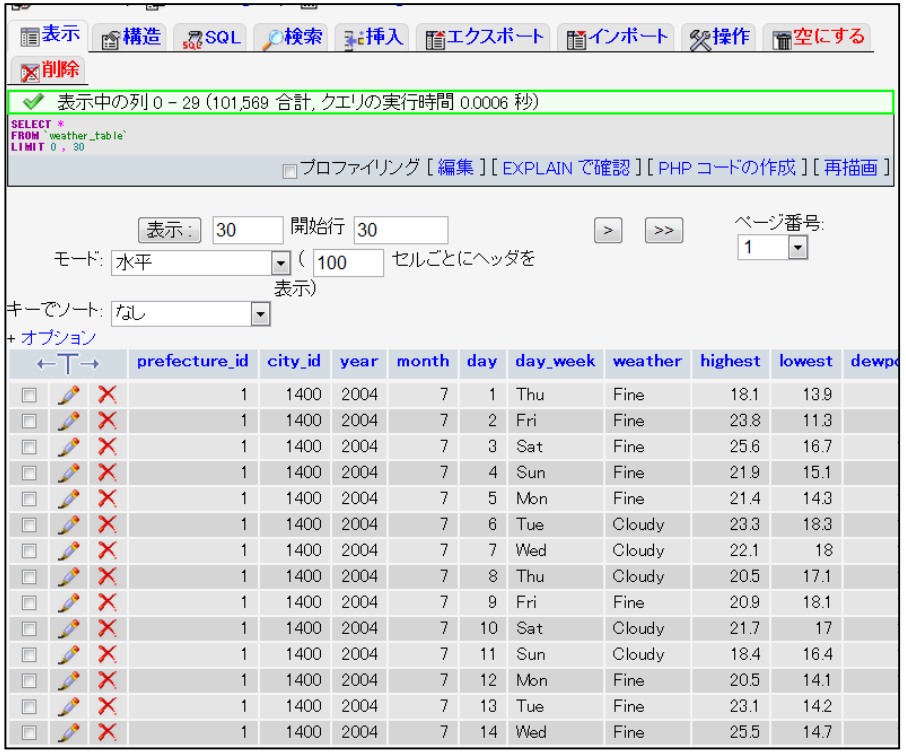
県	年	月	日	曜日	天気	最高気温	最低気温	湿度
京都	2006	11	24	金曜日	晴れ	15.5	10.7	38
京都	2006	11	25	土曜日	曇	15.1	5.1	57
京都	2006	11	26	日曜日	曇	16.1	12.5	63
京都	2006	11	27	月曜日	雨	16.3	14.3	86
京都	2006	11	28	火曜日	雨	18.4	13.5	53

prefecture_id	year	month	day	weekday	weather	highest	lowest	humidity
26	2006	11	24	Fri	Fine	15.5	10.7	38
26	2006	11	25	Sat	Cloudy	15.1	5.1	57
26	2006	11	26	Sun	Cloudy	16.1	12.5	63
26	2006	11	27	Mon	Rain	16.3	14.3	86
26	2006	11	28	Tue	Rain	18.4	13.5	53

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

weather_tableで遊ぼう

- どのようなデータが入っているか確認しましょう
- 2005年10月10日の全国の天気は？
- 2005年の東京（pref=13）の晴れの日の数は？
- 最高気温，最低気温
- 湿度などなど
- 1日の寒暖の差は？



prefecture_id	city_id	year	month	day	day_week	weather	highest	lowest	dewpo
1	1400	2004	7	1	Thu	Fine	18.1	13.9	
1	1400	2004	7	2	Fri	Fine	23.8	11.3	
1	1400	2004	7	3	Sat	Fine	25.6	16.7	
1	1400	2004	7	4	Sun	Fine	21.9	15.1	
1	1400	2004	7	5	Mon	Fine	21.4	14.3	
1	1400	2004	7	6	Tue	Cloudy	23.3	18.3	
1	1400	2004	7	7	Wed	Cloudy	22.1	18	
1	1400	2004	7	8	Thu	Cloudy	20.5	17.1	
1	1400	2004	7	9	Fri	Fine	20.9	18.1	
1	1400	2004	7	10	Sat	Cloudy	21.7	17	
1	1400	2004	7	11	Sun	Cloudy	18.4	16.4	
1	1400	2004	7	12	Mon	Fine	20.5	14.1	
1	1400	2004	7	13	Tue	Fine	23.1	14.2	
1	1400	2004	7	14	Wed	Fine	25.5	14.7	

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

SQLに下記を入力してみましょう

```
SELECT  AVG(highest)  FROM  weather_table;
SELECT  MAX(highest)  FROM  weather_table;
SELECT  MIN(lowest)  FROM  weather_table;
SELECT  AVG(highest-lowest)  FROM  weather_table;
SELECT * FROM weather_table
    where year=2005 AND month=10 AND day=10;
SELECT  COUNT(*)  FROM weather_table
    where prefecture_id=13 AND year=2005;
```

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

SELECT

SELECT 表示属性　　＊の場合は全部

FROM 表名

WHERE 表示条件　　および結合条件

GROUP BY まとめる属性

ORDER BY 並べる属性 並べ方　　ASC / DESC

LIMIT 表示件数　　10だと10件

COUNT()は数を数え，AVG()は平均，MAX()は最大値，MIN()は最小値を計算して返す

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

射影 (projection): $\pi_{\beta}(A)$

顧客NO	名前	年齢	購買した商品
001001	中村 聡史	33	地鶏もも肉
001001	中村 聡史	33	ブルーチーズ
001001	中村 聡史	33	フラン
001002	浅野 泰仁	34	烏龍茶
001002	浅野 泰仁	34	惣菜弁
001003	田中 克己	58	食パン

SELECT 顧客NO, 名前, 購買した商品 FROM TABLEA

顧客NO	名前	購買した商品
001001	中村 聡史	地鶏もも肉
001001	中村 聡史	ブルーチーズ
001001	中村 聡史	フランス産赤ワイン
001002	浅野 泰仁	烏龍茶
001002	浅野 泰仁	惣菜弁当
001003	田中 克己	食パン

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

制限 (selection): $\sigma_{\phi}(A)$

顧客NO	購買した商品
001001	地鶏もも肉
001001	ブルーチーズ
001001	フランス産赤ワイン
001002	烏龍茶
001002	惣菜弁当
001003	食パン

```
SELECT * FROM TABLEA
WHERE 顧客NO = “001001”
```

顧客NO	購買した商品
001001	地鶏もも肉
001001	ブルーチーズ
001001	フランス産赤ワイン

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

表示条件（WHERE）

- 等しい

  weather = "Fine"

  year = 2011
- 未満，より大きい

  highest < 18

  lowest > 0
- 以上，以下

  highest <= 18

  lowest >= 0
- 部分一致

  message like '%naka%'
- かつ

  AND

  month=10 AND day=5
- または

  OR

  year=2010 OR year=2011

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

まとめる（GROUP BY）

GROUP BY まとめる属性

1つの属性でまとめる: 天気ごとにまとめる

```
SELECT weather, count(*)
FROM weather_table
GROUP BY weather;
```

2つの属性でまとめる: 年と月でまとめる

```
SELECT year, month, AVG(highest)
FROM weather_table
GROUP BY year, month;
```

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

ならべる（ORDER BY）

ORDER BY 並べる属性 並べる順序

降順（DESC）：最高気温が高い順に並べる

```
SELECT year, month, day, highest
FROM weather_table
ORDER BY highest DESC;
```

昇順（ASC）：最低気温が低い順に並べる

```
SELECT year, month, day, lowest
FROM weather_table
ORDER BY lowest ASC;
```

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

SELECTで使える集約関数

- 数える（COUNT）
- 平均（AVG）
- 合計（SUM）
- 最大値（MAX）
- 最小値（MIN）

などなど

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

数える（COUNT）

COUNT(数える属性)

総テーブル数

```
SELECT  COUNT(*)  FROM weather_table;
```

東京で2008年で0度以下の日の数

```
SELECT  COUNT(*)  FROM weather_table
WHERE  lowest <= 0 and prefecture_id=13 and year=2008;
```

東京で2007年で晴れた日の数

```
SELECT  COUNT(*)  FROM weather_table
WHERE  weather = "Fine" and prefecture_id=13 and year=2007;
```

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

合計，平均と最大最小

SUM(合計を出したい属性)

AVG(平均を出したい属性)

MAX(最大値を出したい属性)

MIN(最小値を出したい属性)

SELECT **SUM(PRICE)** FROM cart_table;

SELECT **AVG(humidity)** FROM weather_table;

SELECT **MAX(highest)** FROM weather_table;

SELECT **MIN(lowest)** FROM weather_table;

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

演習

- 色々と天気データベースを使いましょう
  - 北海道（pref=1）と，東京（pref=13），沖縄（pref=47）の，最高気温平均，最低気温平均をそれぞれ計算して比較してみましょう
  - 11月の最高・最低気温平均の比較もしてみましょう
  - また，県毎に計算して並び替えてみましょう
  - 京都の月ごとの最高気温平均を求めてみましょう
  - 晴れの日が最も多い県を調べてみましょう
  - 雨の日が最も多い件を調べてみましょう
  - 最も気温が低い，高い県と年月日を調べましょう

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

正規化と正規形

- 正規化とは冗長性を排除するためにある表を複数の表に分解していくこと

- 正規形のタイプ
  - 非正規系（整理されてないデータ）
  - 第1正規形（シンプルな表形式データ）
  - 第2正規形（冗長性をある程度排除）
  - 第3正規形（冗長性がほとんど排除）

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

非正規形

- まったく整理されていない表
- 演算を行うことが出来ない

伝票ID	日付	顧客ID	顧客名	住所	商品ID	単価	数	金額	合計
0010001	2010/5/28	1001	中村 聡史	長崎	ワイン チーズ	1000 500	5 3	5000 1500	6500
0010002	2010/5/29	1002	阪大 太郎	大阪	スナック	350	4	1400	1400
0010003	2010/5/31	1003	京大 花子	京都	ワイン スナック 日本酒	1000 350 2500	5 10 8	5000 3500 20000	28500
0010004	2010/5/31	1001	中村 聡史	長崎	ワイン スナック	1000 350	10 3	10000 1050	1150

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

第1正規形

- キーを設定
- 冗長な部分をカット
  - 繰り返し部分を別表に
  - 導出可能なもの
- データを整理して分割

伝票番号	商品ID	単価	数
0010001	ワイン	1000	5
0010001	チーズ	500	3
0010002	スナック	350	4
0010003	ワイン	1000	5
0010003	スナック	350	10
0010003	日本酒	2500	8
0010004	ワイン	1000	10
0010004	スナック	350	3

伝票ID	日付	顧客ID	顧客名	住所
0010001	2010/5/28	1001	中村 聡史	長崎
0010002	2010/5/29	1002	阪大 太郎	大阪
0010003	2010/5/31	1003	京大 花子	京都
0010004	2010/5/31	1001	中村 聡史	長崎

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

第2正規形

- 冗長な部分をカット
  - 部分関数従属が無くなったもの
  - 非キー属性が，各候補キーに完全関数従属

伝票ID	日付	顧客ID	顧客名	住所
0010001	2010/5/28	1001	中村 聡史	長崎
0010002	2010/5/29	1002	阪大 太郎	大阪
0010003	2010/5/31	1003	京大 花子	京都
0010004	2010/5/31	1001	中村 聡史	長崎

商品ID	単価
ワイン	1000
チーズ	500
スナック	350
日本酒	2500

伝票番号	商品ID	数
0010001	ワイン	5
0010001	チーズ	3
0010002	スナック	4
0010003	ワイン	5
0010003	スナック	10
0010003	日本酒	8
0010004	ワイン	10
0010004	スナック	3

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

第3正規形

- さらに冗長な部分をカット
  - すべての非キー属性が，候補キーから推移関数従属していないこと

伝票ID	日付	顧客ID
0010001	2010/5/28	1001
0010002	2010/5/29	1002
0010003	2010/5/31	1003
0010004	2010/5/31	1001

顧客ID	顧客名	住所
1001	中村 聡史	長崎
1002	阪大 太郎	大阪
1003	京大 花子	京都

伝票ID	商品ID	数
0010001	ワイン	5
0010001	チーズ	3
0010002	スナック	4
0010003	ワイン	5
0010003	スナック	10
0010003	日本酒	8
0010004	ワイン	10
0010004	スナック	3

商品ID	単価
ワイン	1000
チーズ	500
スナック	350
日本酒	2500

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

演習

- 下記の非正規形の表を第1正規形，第2正規形，第3正規形にしてみましょう

Login ID	Login 期間	ID	NAME	所属	File	Printer	Room	枚数	時間
10001	10:00 - 10:35	1001	Satoshi	工学	help.pdf intro.pdf	P01 P01	201	15 20	35
10002	11:37 - 11:42	1002	Taro	医学	help.pdf	P02	202	15	5
10003	12:00 - 12:53	1003	Hanako	理学	help.pdf hanako.pdf report5.pdf	P01 P01 P01	201	15 100 58	53
10004	13:00 - 13:20	1004	Jiro	工学	guide.pdf intro.pdf	P02 P02	202	12 20	20
10005	13:35 - 13:50	1002	Taro	医学	intro.pdf	P02	202	20	15
10006	13:55 - 14:20	1001	Satoshi	工学	guide.pdf	P01	201	12	25

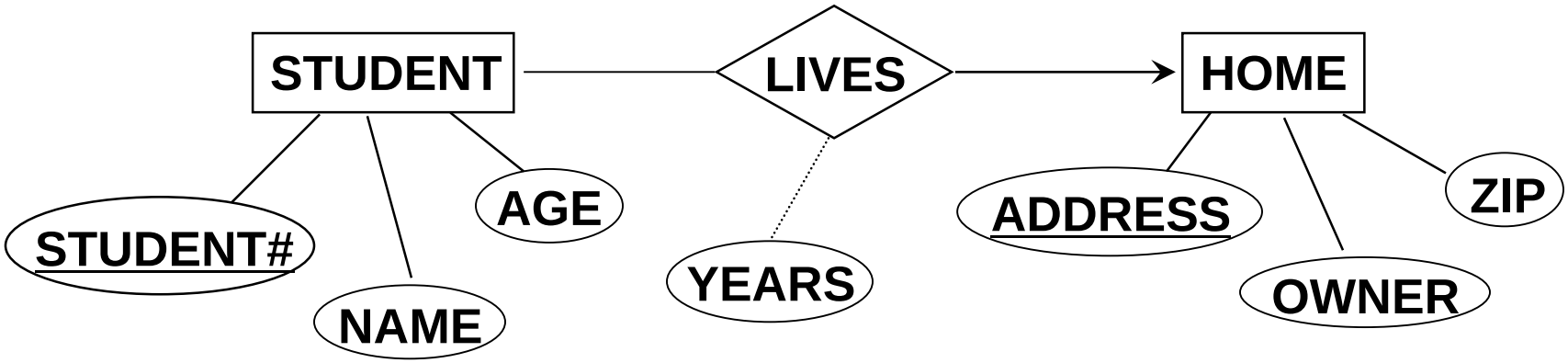
明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

ERモデル

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

ER図を用いたデータ設計

- Entity Relationship Model
  - 実体（実在）の関係モデル化
  - RDBMSの設計に利用される



明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

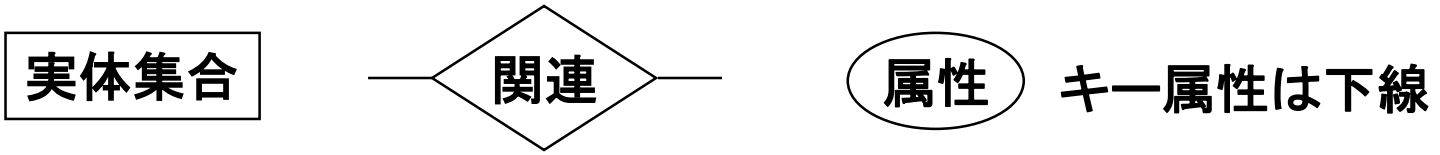
Entity Relationship Model

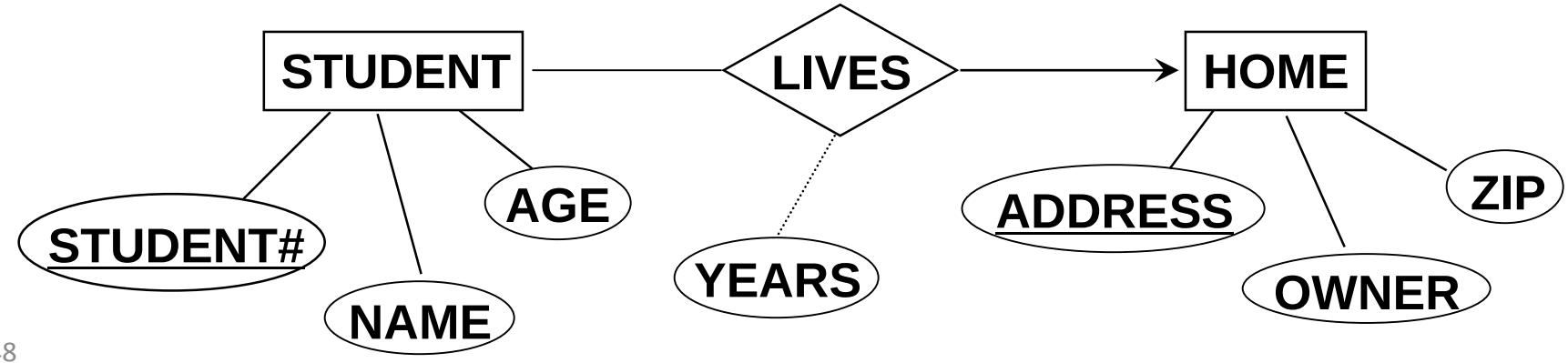
- エンティティ（実体）
  - 実在し，区別可能なもの（顧客，車，DVD，植物など）
  - 抽象的な概念でも良い（資本主義，社会主義など）
- アトリビュート（属性）
  - 実体のもつ特性や付随的な要素（氏名，住所，電話番号等）
  - 属性値は，整数，実数，文字列などの定義域をとる
- リレーションシップ（関連）
  - 実体と実体との間の相互関係（配属，所有，操縦可能など）
  - 一対一（AならBと，AはB），一対多，多対多などの関係
- キー
  - この属性の値により実体集合中の1つの実体を一意に識別

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

Entity Relationship Diagram

- データの構造や関係を視覚化
  - エンティティを四角形，エンティティの属性を楕円，エンティティ同士の関係を菱形で表現
  - エンティティ間を直線で結び，その間にリレーションシップを記述することで人目で把握できるように





明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

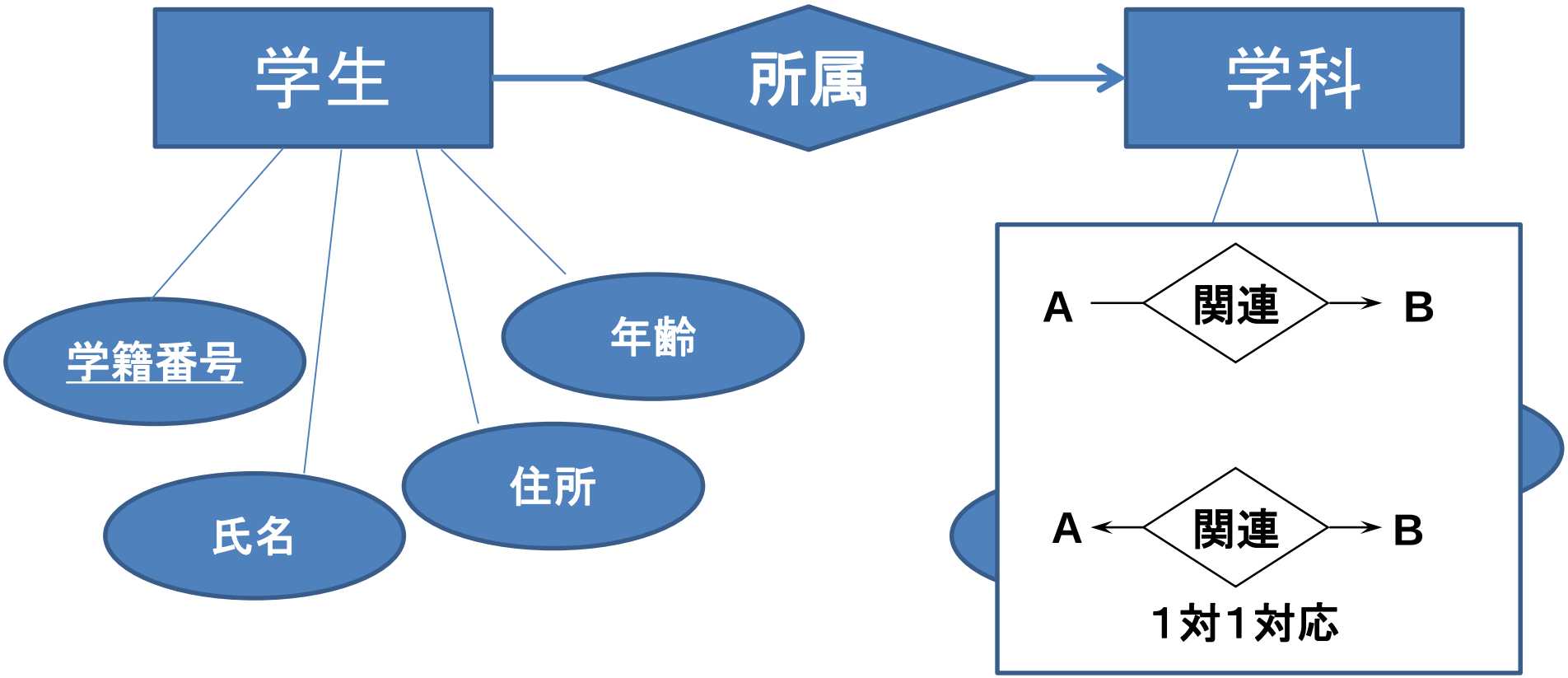
Entity Relationship Model

- 学生と学科の関係は？
  - 学生(氏名，学籍番号，年齢，住所，所属学科)
  - 学科(学科名，住所)

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

Entity Relationship Diagram

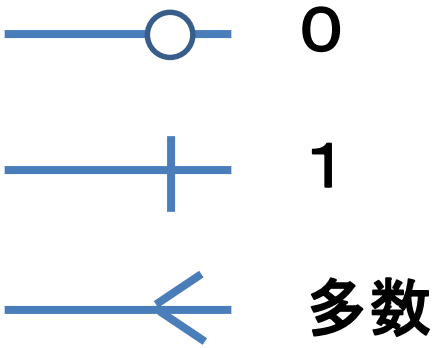
- 学生と学科の関係



オリジナルのERダイアグラム表記法（Peter Chen 記法, 1976）

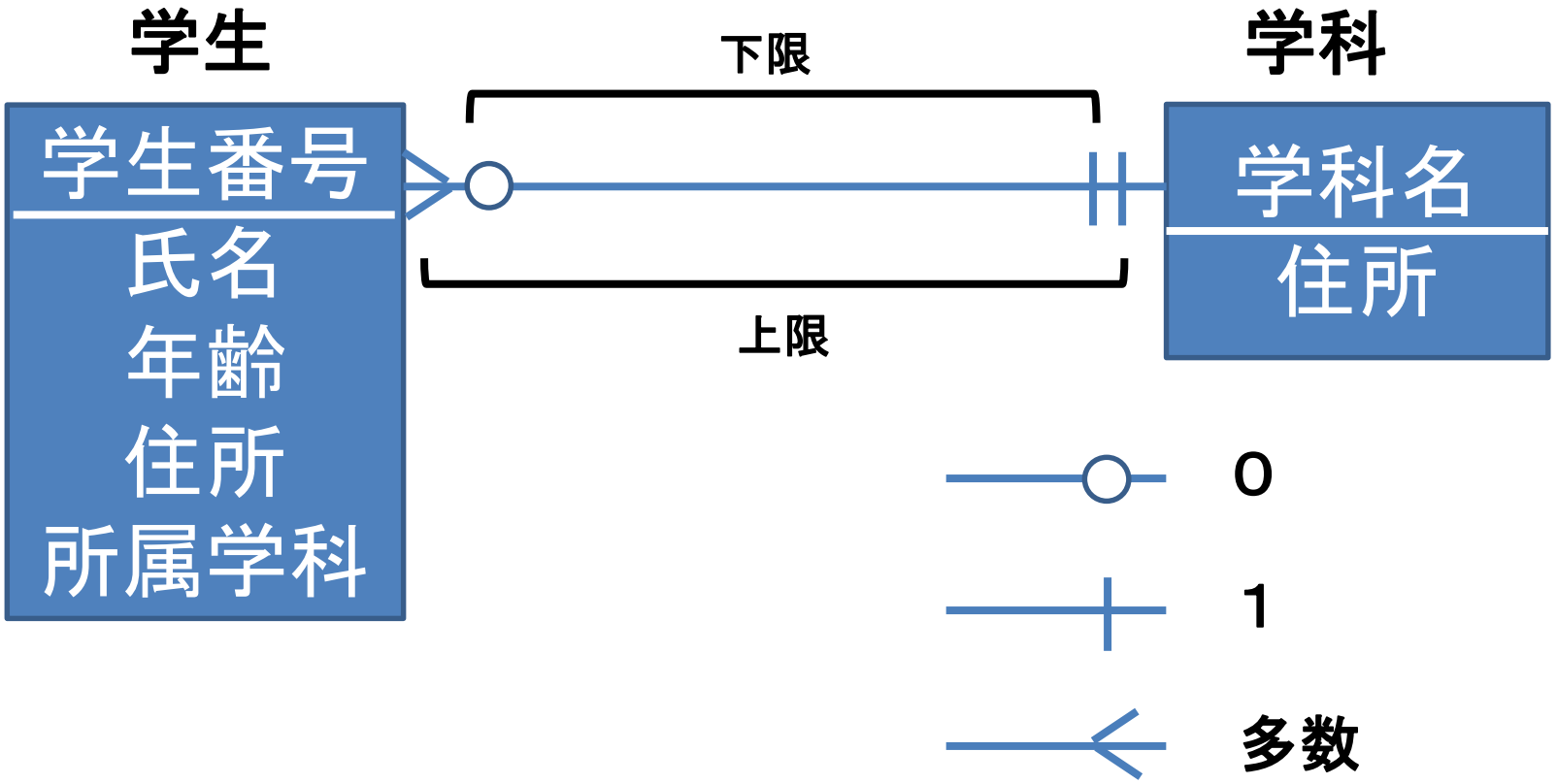
明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

Information Engineering 表記法



明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

Information Engineering 表記法



明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

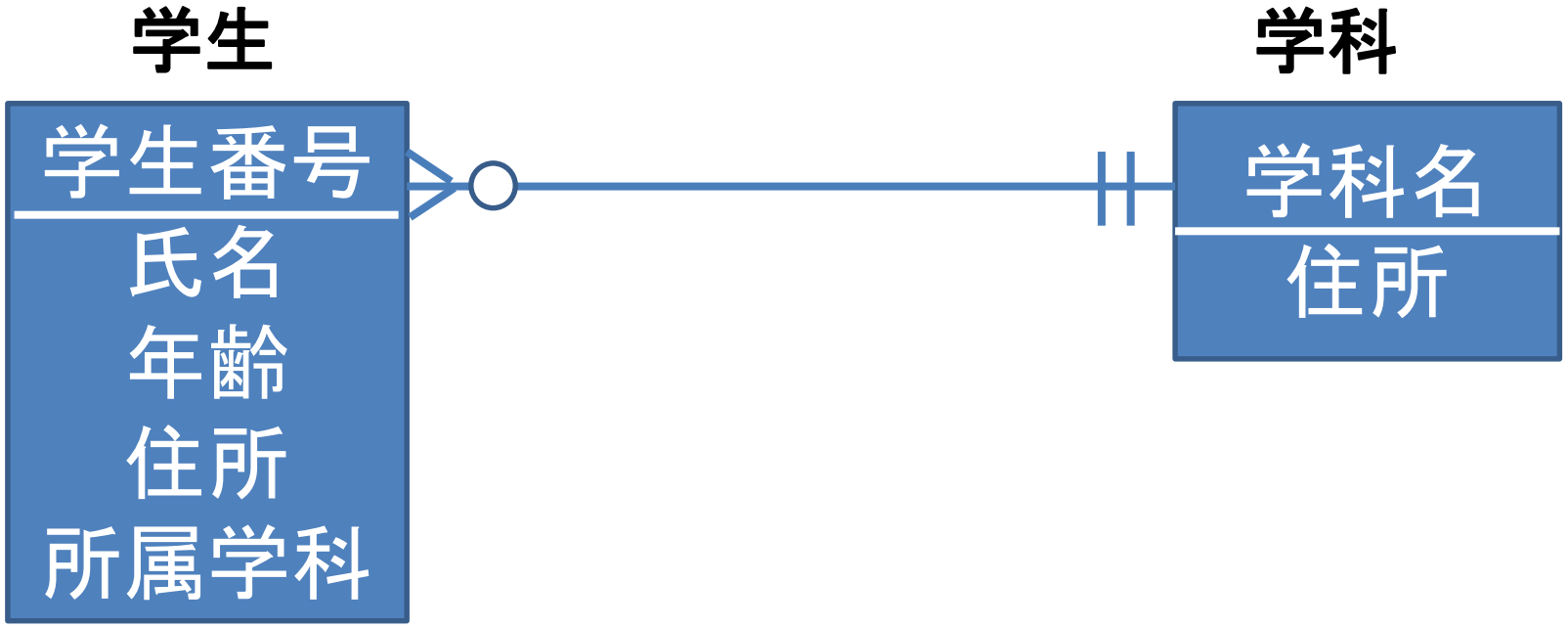
ERとRDB

- ERはRDBに置き換え可能

ER	RDB
エンティティ	テーブル
リレーション	参照における制約
アトリビュート	列名（属性名）
キー	候補キー

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

ER → RDBMS



学生テーブル

学生番号	氏名	年齢	住所	所属学科名

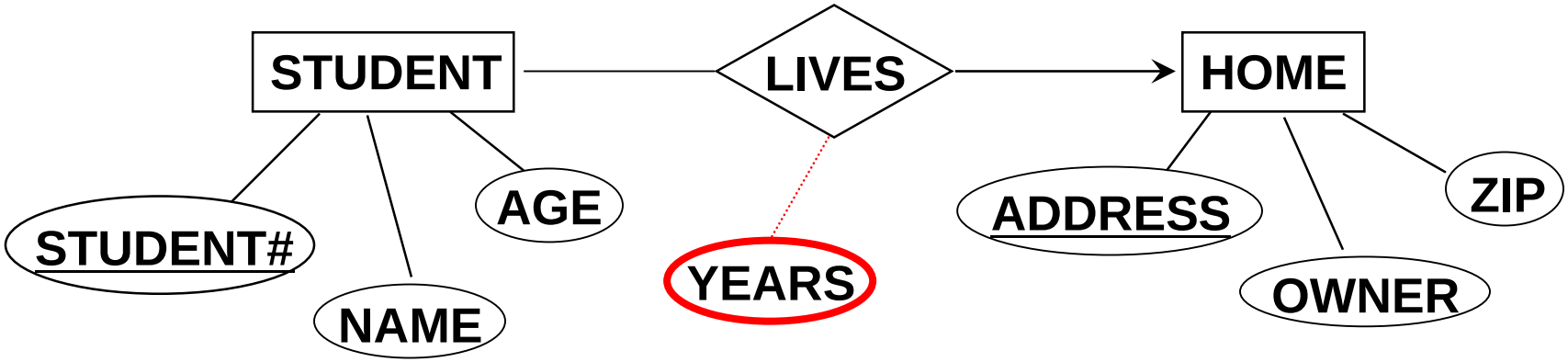
学科テーブル

学科名	住所

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

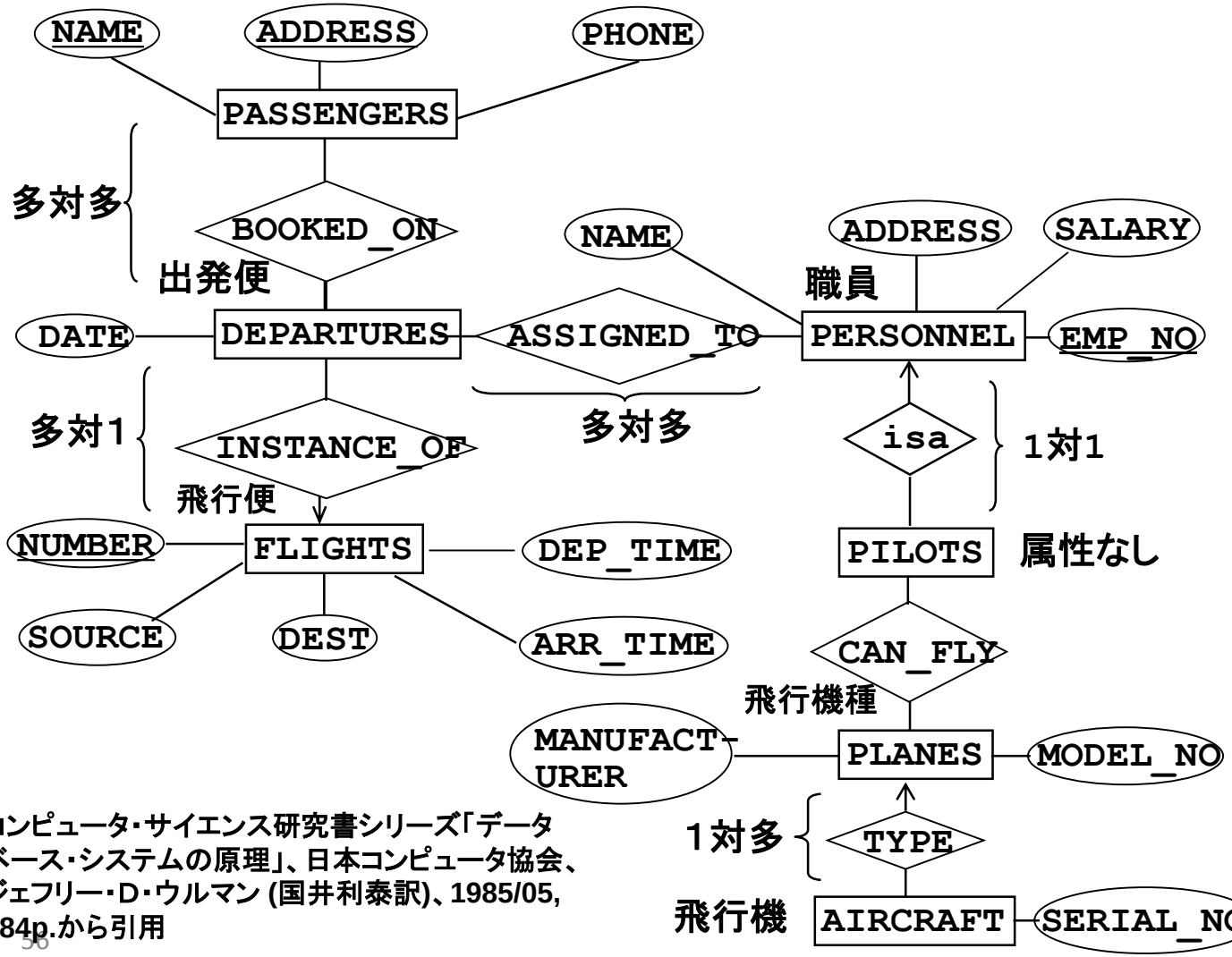
実体関連図の例

- 元のERモデルでは関連にも属性を付加可能



明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

参考（航空会社DBのER）



PASSENGER:乗客
NAME:名前
ADDRESS:住所
PHONE:電話番号
BOOKED_ON:予約
DEPARTURES:出発便
DATE:日付け
INSTANCE:インスタンス
FLIGHTS:飛行便
NUMBER:便番号
SOURCE:出発地
DEST:目的地
DEP_TIME:出発時間
ARR_TIME:到着時間
ASSIGNED_TO:割り当て
PERSONNEL:乗員

SALARY:給料
EMP_NO:従業員番号
PILOTS:操縦士
CAN_FLY:操縦可能
PLANES:飛行
MANUFACTURER:製造会社
MODEL_NO:型番号
TYPE:型
AIRCRAFT:飛行機
SERIAL_NO:通し番号

DEPARTURESの各実体は、FLIGHTのNUMBERとDEPARTUREのDATEによって一意に識別。

コンピュータ・サイエンス研究書シリーズ「データベース・システムの原理」、日本コンピュータ協会、ジェフリー・D・ウルマン (国井利泰訳)、1985/05, 584p.から引用

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

ER図を作ってみましょう

- ワールドカップの登録選手について
  - 参加する国は名前と地域がある，その国の代表チームには，監督と選手が所属している．監督は名前と年俸，年齢をもつ．サッカー選手は名前，年俸，年齢，ポジション，所属クラブチームの情報をもつ．
  - 各国は8つのいずれかのグループリーグに所属している．
  - 試合日程は，試合日，試合会場，決まっている場合はその対戦する両チームからなる．試合結果の記録は，試合日，試合会場，対戦する両チームの情報，スコアからなる．

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

2つの表をどうくっつける？

- 複数の表に分割したほうが良い
- どうやって複数の表の情報を統合するのか？

顧客NO	名前	年齢
001001	中村 聡史	33
001002	浅野 泰仁	34
001003	田中 克己	58

顧客NO	購買した商品
001001	地鶏もも肉
001001	ブルーチーズ
001001	フランス産赤ワイン
001002	烏龍茶
001002	惣菜弁当
001003	食パン

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

結合 (join): A⋈B

顧客NO	名前	年齢
001001	中村 聡史	33
001002	浅野 泰仁	34
001003	田中 克己	58

顧客NO	購買した商品
001001	地鶏もも肉
001001	ブルーチーズ
001001	フランス産赤ワイン
001002	烏龍茶
001002	惣菜弁当
001003	食パン

顧客NO	名前	年齢	購買した商品
001001	中村 聡史	33	地鶏もも肉
001001			
001001			
001002			
001002			
001003			

```
SELECT  TABLEA.顧客NO, TABLEB.名前,
        TABLEA.年齢, TABLEB.購買した商品
FROM  TABLEA, TABLEB
WHERE TABLEA.顧客NO = TABLEB.顧客NO
```

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

SELECT（表をつなげる）

```
SELECT
   area_table.city_name,
   avg(weather_table.highest)
FROM  weather_table, area_table
WHERE
   weather_table.city_id = area_table.city_id
GROUP  BY  weather_table.city_id;
```

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

内部結合と外部結合

- 内部結合とは，それぞれの表に該当するもののみが抽出されてテーブルとして作られる
  - INNER JOIN
- 外部結合とは，どちらかの表は完全に読み込み，他方の表については該当部分についてのみ結合されるというもの
  - LEFT OUTER JOIN
  - RIGHT OUTER JOIN

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

内部結合と外部結合

顧客

ID	NAME
1	三輪 聡哉
2	和田 拓哉
3	菅澤 卓也

伝票

伝票ID	購入者ID	支払
1	1	15000
2	3	8000
3	1	7800

ID	NAME	伝票ID	支払
1	三輪 聡哉	1	15000
3	菅澤 卓也	2	8000
1	三輪 聡哉	3	7800

内部結合 INNER JOIN

```
SELECT 顧客.ID, 顧客.名前, 伝票.伝票ID, 伝票.支払
FROM 顧客, 伝票
WHERE 顧客.ID = 伝票.購入者ID
```

```
SELECT 顧客.ID, 顧客.名前, 伝票.伝票ID, 伝票.支払
FROM 顧客 INNER JOIN 伝票
ON 顧客.ID = 伝票.購入者ID
```

ID	NAME	伝票ID	支払
1	三輪 聡哉	1	15000
1	三輪 聡哉	3	7800
2	和田 拓哉		
3	菅澤 卓也	2	8000

外部結合 LEFT OUTER JOIN

```
SELECT 顧客.ID, 顧客.名前, 伝票.伝票ID, 伝票.支払
FROM 顧客 LEFT JOIN 伝票
ON 顧客.ID = 伝票.購入者ID
```

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

演習

- area_table と統合して，県IDではなく，県名としてKyotoを指定し，2009年6月28日の天気と最高気温，最低気温を出力するSQLを作成せよ
- Hokkaidoの月毎の最高気温，最低気温，湿度の平均を出力するSQLを作成せよ．Okinawaについても同様に試してみよ

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

SQL

- SQL と呼ばれるクエリを利用して処理を行う

CREATE（データベースやテーブルを作る）

DROP（データベースやテーブルを削除する）

ALTER（テーブルの定義を変更する）

USE（データベースを選択）

INSERT INTO（テーブルにデータの挿入）

UPDATE ～ SET（テーブルのデータを更新）

DELETE FROM（テーブルからデータの削除）

SELECT ～ FROM ～ WHERE（結果を抽出）

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

CREATE

- CREATE DATABASE データベース名
- CREATE TABLE テーブル名 (テーブルの属性);

（例）

```
CREATE DATABASE CUSTOMER_DB;
CREATE TABLE CUSTOMER
 ( ID int (10) NOT NULL auto_increment,
  NAME char(40),
  AGE’ int(3),
  PRIMARY KEY (ID) );
```

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

DROP

- DROP DATABASE データベース名
- DROP TABLE テーブル名 (テーブルの属性);

（例）

DROP DATABASE CUSTOMER_DB;

DROP TABLE CUSTOMER_TABLE;

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

USE

- USE データベース名
  - 使用するデータベースを選択する
  - 実際にはカタログと呼ばれるところにこれまで作成したデータベースが格納されている．そこからデータベースを選択するコマンド

（例）

- **USE CUSTOMER_DB;**

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

INSERT

INSERT INTO テーブル名 (属性名リスト)
VALUES (内容リスト);

（例）

```
INSERT INTO CUSTOMER_TABLE
    ( ID, NAME, AGE ) VALUES ( 1, '中村聡史', 33 );
```

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

UPDATE

UPDATE テーブル名 SET 変更内容
WHERE 変更条件;

（例）

```
UPDATE CUSTOMER _TABLE
       SET AGE = 34
       WHERE ID = 1;
UPDATE CUSTOMER_TABLE
       SET AGE = 38, NAME = '京大太郎'
       WHERE ID = 2;
```

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

DELETE

DELETE FROM テーブル名 WHERE 削除条件;

（例）

IDが2のカスタマーを削除

```
DELETE FROM CUSTOMER_TABLE
        WHERE ID = 2;
```

年齢が60歳より大きいカスタマーを削除

```
DELETE FROM CUSTOMER_TABLE
        WHERE AGE > 60;
```

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

SELECT

- SELECT 表示属性 FROM テーブル名
  WHERE 表示条件
  ORDER BY 並べる列;

（例）

```
SELECT NAME, AGE FROM CUSTOMER_TABLE
    WHERE ID = 1;
SELECT NAME, AGE FROM CUSTOMER_TABLE
    WHERE AGE < 40 AND AGE > 30
    ORDER BY AGE;
```

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

演習: 家計簿データベースを作ろう

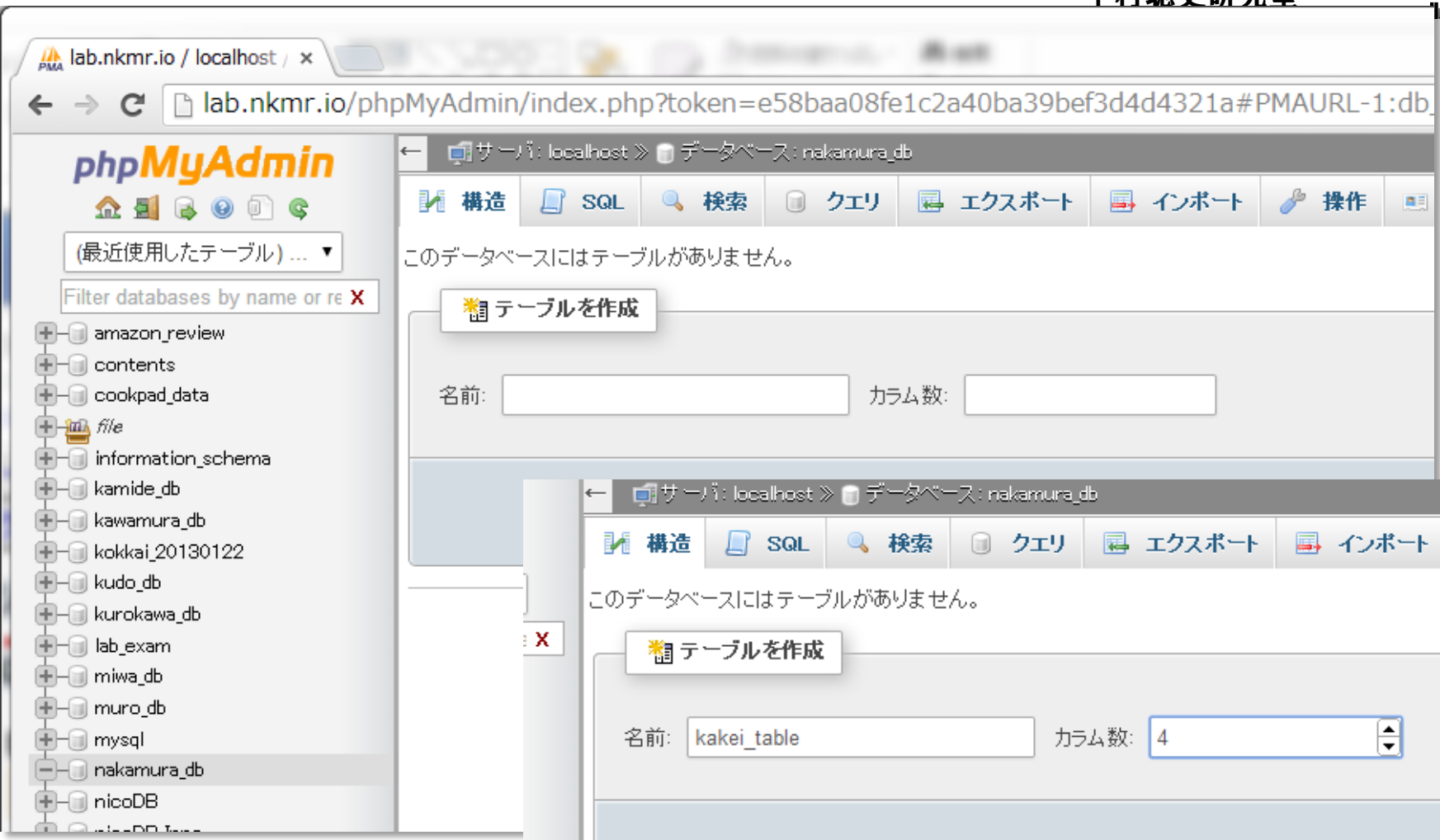
内容は各自適当に

ID	YEAR	MONTH	DAY	CONTENT	PAYMENT
1	2012	11	26	家賃	50000
2	2012	11	28	ラーメン	650
3	2012	11	29	本	2500

手順

1. ユーザ名_db（自分のっぽいのを探して下さい）というデータベースを選択し，kakei_table を作成
2. データの挿入

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

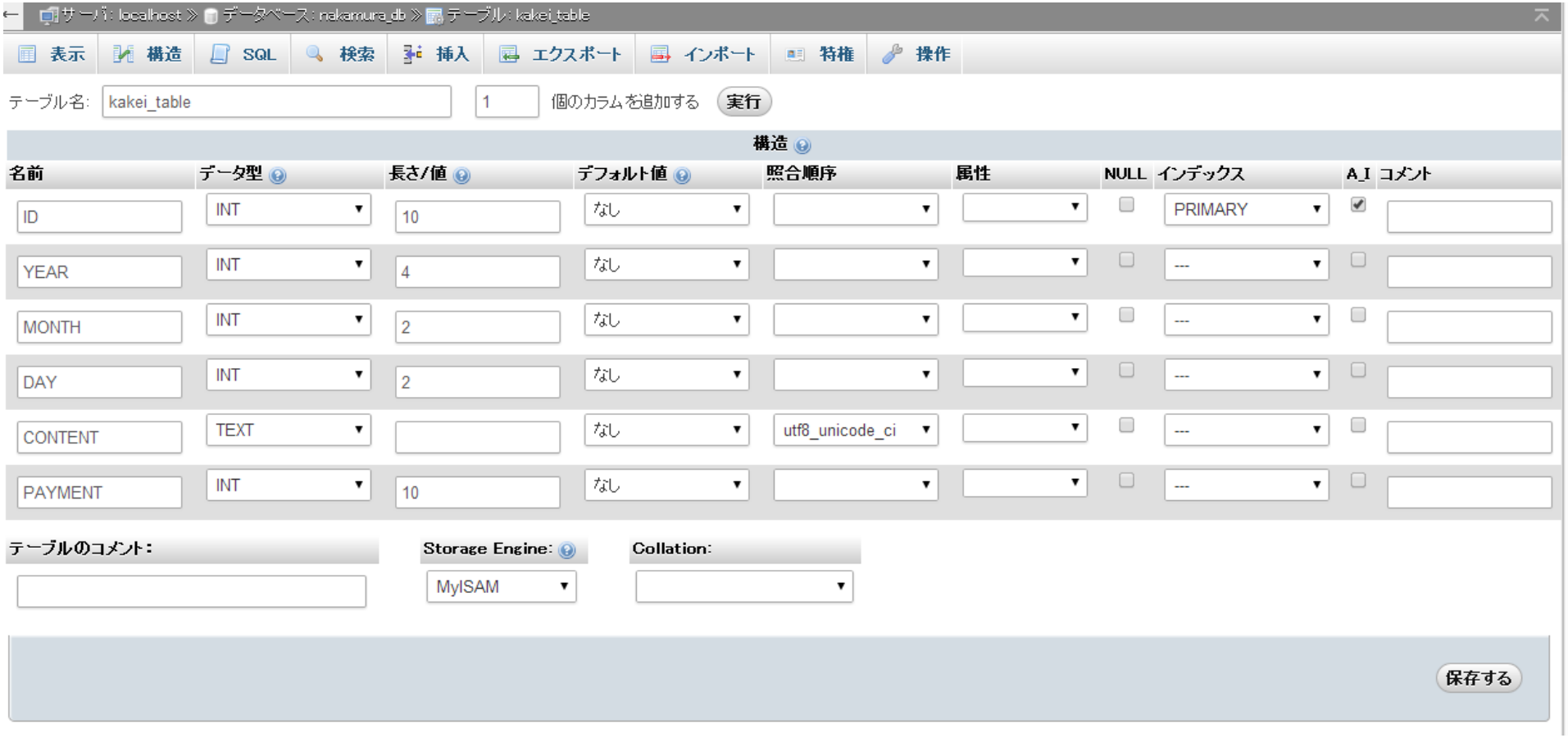


明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

どんなテーブルにするか...

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

テーブルの中身を定義



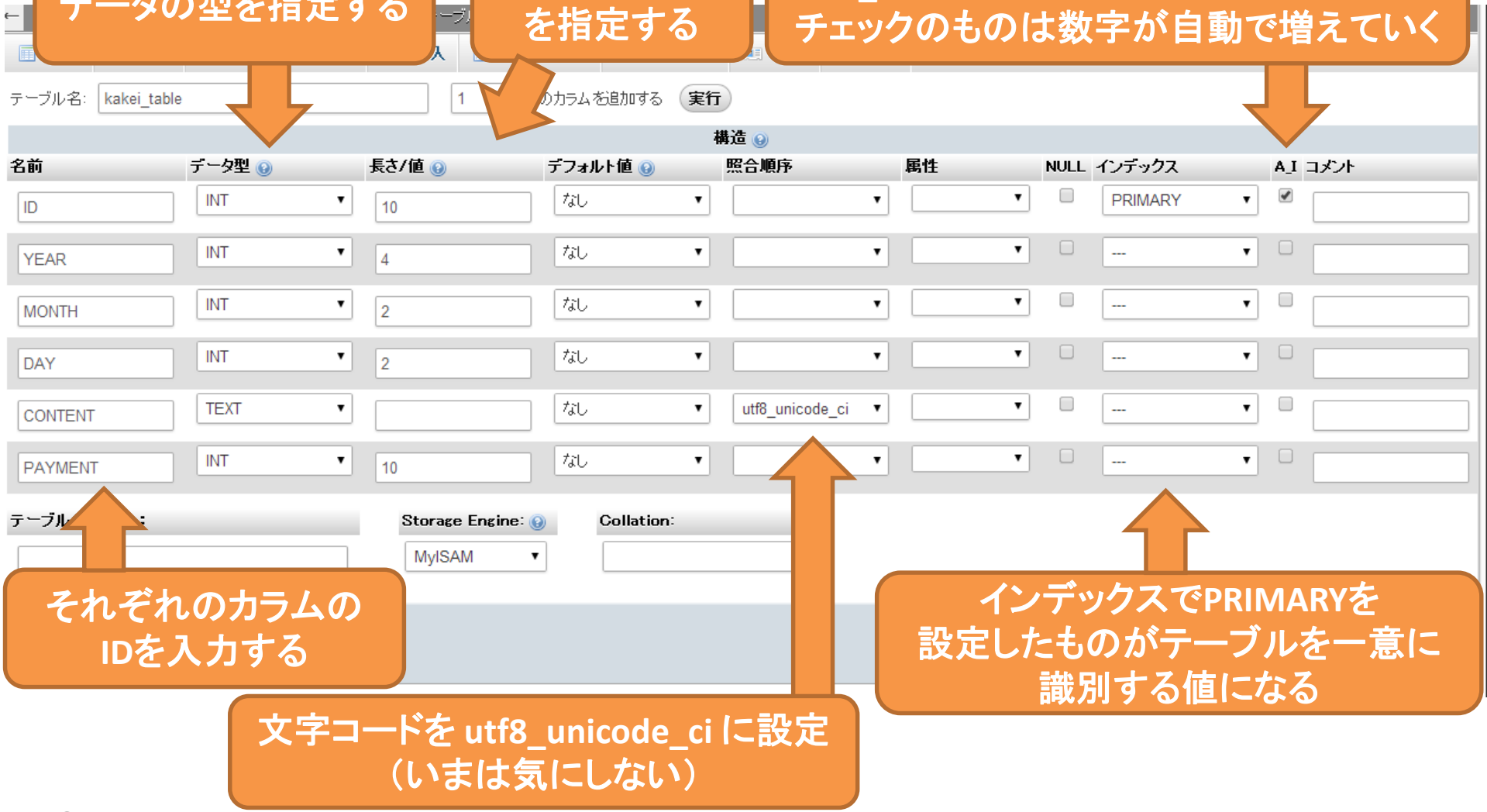
明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

テーブルの中身を定義

データの型を指定する

整数等の桁数を指定する

A_I はオートインクリメントという意味
チェックのものは数字が自動で増えていく



それぞれのカラムの
IDを入力する

文字コードを utf8_unicode_ci に設定
（いまは気にしない）

インデックスでPRIMARYを
設定したものがテーブルを一意に
識別する値になる

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

入力してみましょう


```
INSERT INTO
kakei_table (ID, YEAR, MONTH, DAY, CONTENT, PAYMENT)
VALUES (1, 2014, 4, 11, "定食", 480 );
```

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

データを挿入してみる（SQL）

2014年4月10日にラーメン650円を購入

INSERT INTO kakei_table (ID, YEAR, MONTH, DAY, CONTENT, PAYMENT)

VALUES (1, 2014, 4, 10, "ラーメン", 650);

2014年4月11日に学食で480円の定食を頼む

INSERT INTO kakei_table (ID, YEAR, MONTH, DAY, CONTENT, PAYMENT)

VALUES (2, 2014, 4, 11, "学食", 480);

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

データを挿入してみる（SQL）

2014年4月10日にラーメン650円を購入

```
INSERT INTO kakei_table (YEAR, MONTH, DAY,
CONTENT, PAYMENT)
VALUES (2014, 4, 10, "ラーメン", 650);
```

Auto Increment（A_I）のものは省略可能

2014年4月11日に学食で480円の定食を頼む

```
INSERT INTO kakei_table (YEAR, MONTH, DAY,
CONTENT, PAYMENT)
VALUES (2014, 4, 11, "学食", 480);
```

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

家計状況の確認

全支払いチェック

```
SELECT * FROM kakei_table;
```

4月の全支払いチェック

```
SELECT * FROM kakei_table
WHERE MONTH = 4;
```

2014年の総支払額確認

```
SELECT SUM(PAYMENT) FROM kakei_table
WHERE YEAR = 2014;
```

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

情報の変更，追加，削除

支払額の変更（650円から750円に）

```
UPDATE kakei_table SET PAYMENT = 750
WHERE ID = 1;
```

名称の変更（注文したのはラーメン大だった）

```
UPDATE kakei_table
SET CONTENT="ラーメン大"
WHERE ID = 1;
```

支払い情報の削除（学食のは削除）

```
DELETE FROM kakei_table WHERE ID = 2;
```

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

演習

- 支払情報をどんどん追加してみましょう
- 日毎の総額を表示してみましょう
- 項目を追加してみましょう
  - 食費，娯楽費，交際費，家賃などなど
- 収入を追加してみましょう

- 項目テーブルを別途作成してみましょう
  - 項目ID，項目名など

明治大学総合数理学部
先端メディアサイエンス学科
中村聡史研究室

宿題

- 県ごとの最高気温と最低気温の差の平均を出力するSQLを作成せよ．なお，差が少ない順に並べよ．また，県名を同時に表示せよ
- 2007年で最高気温が30度を超えている日の数の多い県ランキングを作成するSQLと結果を示せ
- 2008年の年間降水量ランキングを作成せよ．表示の際には県名と降水量を示せ
- 2008年の晴れの日が多い県ランキングを作成するSQLと結果を示せ
- 2009年で湿度が60%を超えている日の数の少ない県ランキングを作成するSQLと結果を示せ