

データベースシステム入門

4. リレーショナルデータベースシステムのテーブル, SQL

4-1 SQL の機能紹介

リレーショナルデータベースシステム

コンピュータ



リレーショナル
データベース
管理システム



記憶
装置

リレーショナル
データベース

あわせて
リレーショナルデータベースシステム

	id	name	teacher_name	student_name	score	created_at	updated_at
1	1	Database	K	KK	85	2014-09-13 00:46:17	{null}
2	2	Database	K	AA	75	2014-09-13 00:48:26	{null}
3	3	Database	K	LL	90	2014-09-13 00:48:26	{null}
4	4	Programming	A	KK	85	2014-09-13 00:48:26	{null}
5	5	Programming	A	LL	75	2014-09-13 00:48:26	{null}

	id	year	month	day	customer_name	product_id	qty	created_at
1	1001	2014	10	26	kaneko	1	10	2014-09-14 14:28:38
2	1002	2014	10	26	miyamoto	2	2	2014-09-14 14:28:38
3	1003	2014	10	27	kaneko	3	8	2014-09-14 14:28:38
4	1004	2014	10	27	kaneko	3	8	2014-09-14 14:28:38

データの種類ごとに分かれた、
たくさんの**テーブル**が格納される

欲しいデータを探す，複数のテーブルを1つに結合する = SQL でできる

Duration: 0.038 seconds * Col: 33 Row: 2/2

	id	year	month	day	created_at	id	order_id	qty	shipping_id	created_at	id	year	month	day	customer_name	product_id
1	101	2009	10	28	2014-09-14 14:33:37	1	1001	2	101	2014-09-14 14:33:27	1001	2014	10	26	kaneko	1
2	101	2009	10	28	2014-09-14 14:33:37	1	1001	2	101	2014-09-14 14:33:27	1002	2014	10	26	miyamoto	2
3	101	2009	10	28	2014-09-14 14:33:37	1	1001	2	101	2014-09-14 14:33:27	1003	2014	10	27	kaneko	3
4	101	2009	10	28	2014-09-14 14:33:37	1	1001	2	101	2014-09-14 14:33:27	1004	2014	10	27	kaneko	3
5	101	2009	10	28	2014-09-14 14:33:37	2	1001	1	102	2014-09-14 14:33:27	1001	2014	10	26	kaneko	1
6	101	2009	10	28	2014-09-14 14:33:37	2	1001	1	102	2014-09-14 14:33:27	1002	2014	10	26	miyamoto	2
7	101	2009	10	28	2014-09-14 14:33:37	2	1001	1	102	2014-09-14 14:33:27	1003	2014	10	27	kaneko	3
8	101	2009	10	28	2014-09-14 14:33:37	2	1001	1	102	2014-09-14 14:33:27	1004	2014	10	27	kaneko	3
9	101	2009	10	28	2014-09-14 14:33:37	3	1002	1	103	2014-09-14 14:33:27	1001	2014	10	26	kaneko	1
10	101	2009	10	28	2014-09-14 14:33:37	3	1002	1	103	2014-09-14 14:33:27	1002	2014	10	26	miyamoto	2
11	101	2009	10	28	2014-09-14 14:33:37	3	1002	1	103	2014-09-14 14:33:27	1003	2014	10	27	kaneko	3
12	101	2009	10	28	2014-09-14 14:33:37	3	1002	1	103	2014-09-14 14:33:27	1004	2014	10	27	kaneko	3
13	101	2009	10	28	2014-09-14 14:33:37	4	1003	2	104	2014-09-14 14:33:27	1001	2014	10	26	kaneko	1
14	101	2009	10	28	2014-09-14 14:33:37	4	1003	2	104	2014-09-14 14:33:27	1002	2014	10	26	miyamoto	2
15	101	2009	10	28	2014-09-14 14:33:37	4	1003	2	104	2014-09-14 14:33:27	1003	2014	10	27	kaneko	3
16	101	2009	10	28	2014-09-14 14:33:37	4	1003	2	104	2014-09-14 14:33:27	1004	2014	10	27	kaneko	3
17	101	2009	10	28	2014-09-14 14:33:37	5	1004	4	104	2014-09-14 14:33:27	1001	2014	10	26	kaneko	1
18	101	2009	10	28	2014-09-14 14:33:37	5	1004	4	104	2014-09-14 14:33:27	1002	2014	10	26	miyamoto	2
19	101	2009	10	28	2014-09-14 14:33:37	5	1004	4	104	2014-09-14 14:33:27	1003	2014	10	27	kaneko	3
20	101	2009	10	28	2014-09-14 14:33:37	5	1004	4	104	2014-09-14 14:33:27	1004	2014	10	27	kaneko	3
21	102	2009	10	31	2014-09-14 14:33:37	1	1001	2	101	2014-09-14 14:33:27	1001	2014	10	26	kaneko	1
22	102	2009	10	31	2014-09-14 14:33:37	1	1001	2	101	2014-09-14 14:33:27	1002	2014	10	26	miyamoto	2
23	102	2009	10	31	2014-09-14 14:33:37	1	1001	2	101	2014-09-14 14:33:27	1003	2014	10	27	kaneko	3
24	102	2009	10	31	2014-09-14 14:33:37	1	1001	2	101	2014-09-14 14:33:27	1004	2014	10	27	kaneko	3
25	102	2009	10	31	2014-09-14 14:33:37	2	1001	1	102	2014-09-14 14:33:27	1001	2014	10	26	kaneko	1
26	102	2009	10	31	2014-09-14 14:33:37	2	1001	1	102	2014-09-14 14:33:27	1002	2014	10	26	miyamoto	2
27	102	2009	10	31	2014-09-14 14:33:37	2	1001	1	102	2014-09-14 14:33:27	1003	2014	10	27	kaneko	3
28	102	2009	10	31	2014-09-14 14:33:37	2	1001	1	102	2014-09-14 14:33:27	1004	2014	10	27	kaneko	3
29	102	2009	10	31	2014-09-14 14:33:37	3	1002	1	103	2014-09-14 14:33:27	1001	2014	10	26	kaneko	1

Duration: 0.003 seconds * Col: 1 Row: 6/7

	month	day	qty
1	11	2	2
2	11	2	4

SQLはコマンド言語

1つのテーブルを複数のテーブルに分解したい、 ということもある = SQL でできる

テーブルの分解 (Table Decomposition)

■ 分解前

results(name, teacher_name, student_name, score)

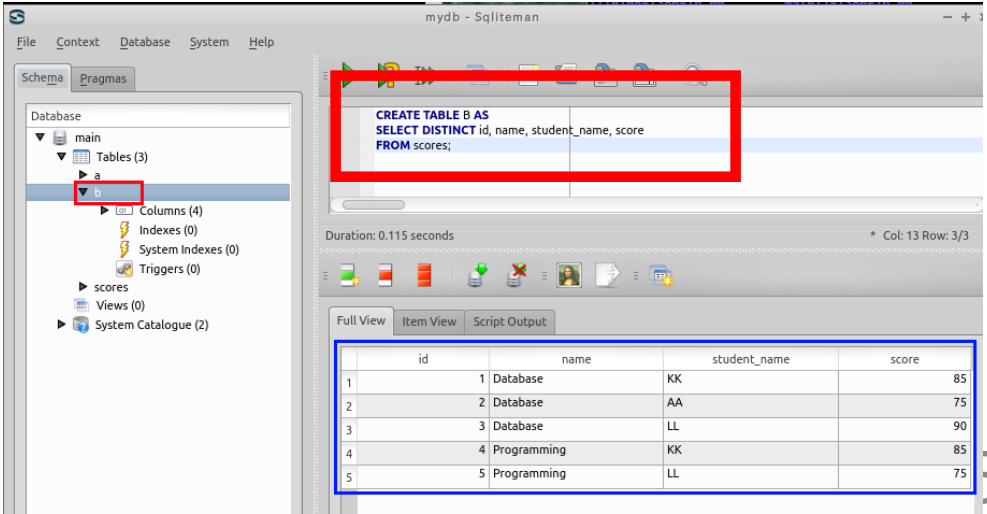
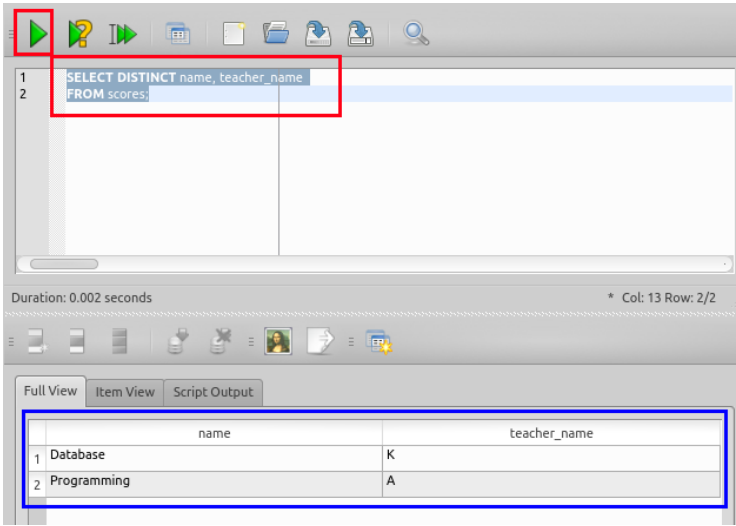
name	teacher_name	student_name	score
Database	K	KK	85
Database	K	AA	75
Database	K	LL	90
Programming	A	KK	85
Programming	A	LL	75

■ 分解後

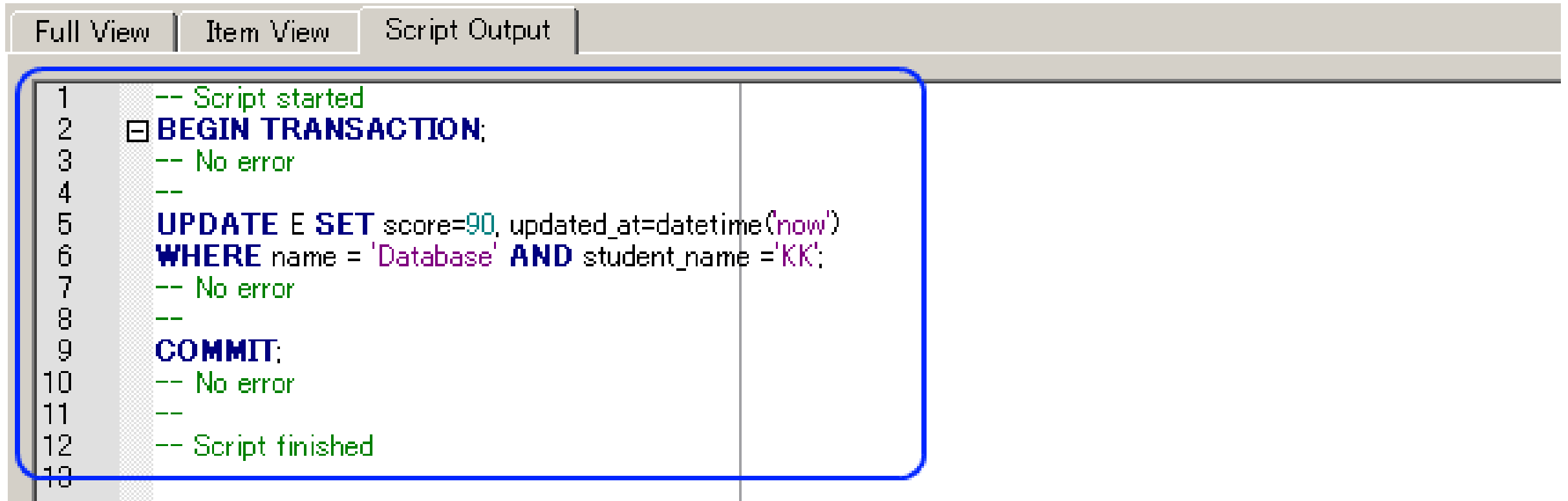
A(name, teacher_name), B(name, student_name, score)

name	teacher_name
Database	K
Programming	A

name	student_name	score
Database	KK	85
Database	AA	75
Database	LL	90
Programming	KK	85
Programming	LL	75



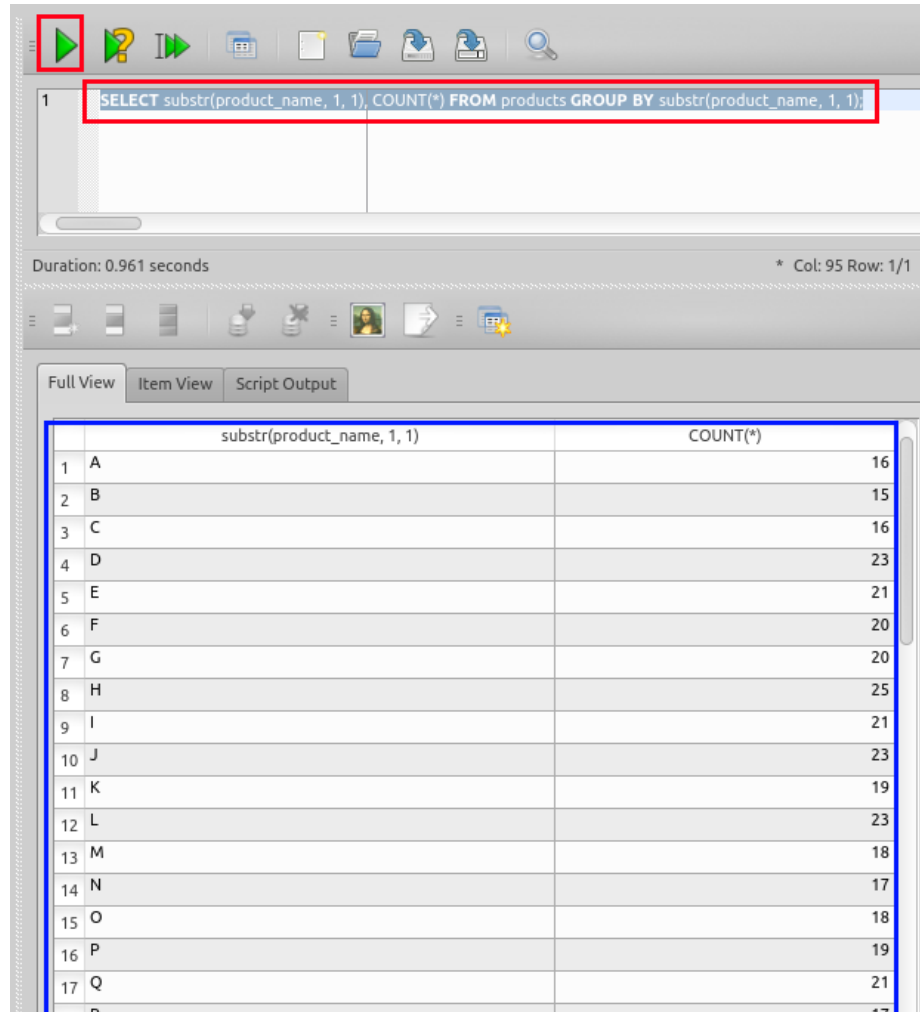
データを最新のものに更新したい, こともある
= SQL でできる

A screenshot of a SQL script editor window. The window has three tabs: 'Full View', 'Item View', and 'Script Output'. The 'Full View' tab is active. The script content is as follows:

```
1  -- Script started
2  BEGIN TRANSACTION;
3  -- No error
4  --
5  UPDATE E SET score=90, updated_at=datetime('now')
6  WHERE name = 'Database' AND student_name = 'KK';
7  -- No error
8  --
9  COMMIT;
10 -- No error
11 --
12 -- Script finished
13
```

The script is enclosed in a blue rounded rectangle. The text is color-coded: comments are green, SQL keywords are blue, and string literals are purple.

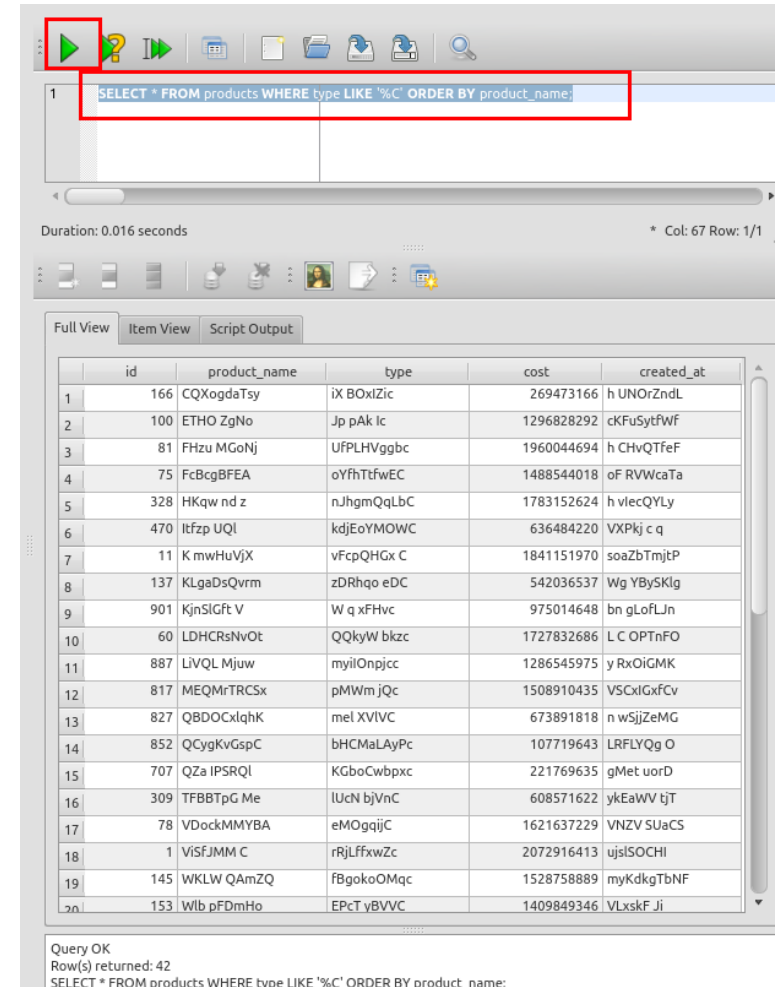
集計，集約したい，ソート（並べ替え）したいこともある = SQL でできる



Duration: 0.961 seconds * Col: 95 Row: 1/1

	substr(product_name, 1, 1)	COUNT(*)
1	A	16
2	B	15
3	C	16
4	D	23
5	E	21
6	F	20
7	G	20
8	H	25
9	I	21
10	J	23
11	K	19
12	L	23
13	M	18
14	N	17
15	O	18
16	P	19
17	Q	21

集計，集約の例



Duration: 0.016 seconds * Col: 67 Row: 1/1

	id	product_name	type	cost	created_at
1	166	CQXogdaTsy	iX BOxlZic	269473166	h UNOrZndL
2	100	ETHO ZgNo	Jp pAk Ic	1296828292	ckFuSytfWf
3	81	FHzu MGoNj	UFPLHVggbc	1960044694	h CHvQTfE
4	75	FcBcgBFEA	oYfhTtfwEC	1488544018	oF RVWcaTa
5	328	HKqw nd z	nJhgmQqLbC	1783152624	h vlecQYLy
6	470	Itfzp UQL	kdjEoYMOWC	636484220	VXPKj c q
7	11	K mwHuVjX	vFcpQHGX C	1841151970	soaZbTmjTP
8	137	KLgaDsQvrm	zDRhgo eDC	542036537	Wg YBySKlg
9	901	KjnSIGft V	W q xFHvc	975014648	bn gLoFLJn
10	60	LDHCRsNvOt	QQkyW bkzc	1727832686	L C OPTnFO
11	887	LIVQL Mjuw	myilOnpjcc	1286545975	y RxOiGMK
12	817	MEQMrTRCSx	pMWm jQc	1508910435	VSCxlGxfCv
13	827	QBDOClqhK	mel XVIVC	673891818	n wSijZeMG
14	852	QCygKvGspC	bHCMaLayPc	107719643	LRFLYQg O
15	707	QZa IPSRQL	KGboCwbpxc	221769635	gMet uorD
16	309	TFBBTpG Me	lUcN bjVnC	608571622	ykEaWV tjT
17	78	VDockMMYBA	eMOgqijC	1621637229	VNZV SUaCS
18	1	ViSFJMM C	rRjLffxwZc	2072916413	ujslSOCHI
19	145	WKLW QAmZQ	fBgokoOMqc	1528758889	myKdkgTbNF
20	153	Wlb pFDmHo	EPcT yBVVC	1409849346	VLxskF Ji

Query OK
Row(s) returned: 42
SELECT * FROM products WHERE type LIKE '%C' ORDER BY product_name;

ソートの例

SQLをマスターするには

◆ SQL のキーワード

create table	テーブル定義
select	射影など
from	扱うテーブルの指定
where	選択, 結合など
max	最大
min	最小
count	数え上げ
group by	集約 (グループの基準の指定)
order by	ソート (並べ替え)
distinct	重複除去
など	

◆ SQL のデータ型

短いテキスト	char
長いテキスト	text
数値	integer, real
日付／時刻	datetime
YesやNo	bit
など	

SQL をマスターしてみませんか？

リレーショナルデータベースシステム

◆ 無料のものも

SQLite3, PostgreSQL, MySQL, Firebird など

◆ あらゆる場所で活躍

Web システム, 専用端末（コンピュータ）との連携システム
オンラインで, データを蓄積, 共有

◆ スマホにも

SQLite3 が内蔵されていることがふつう
(Webブラウザの履歴管理などのため)

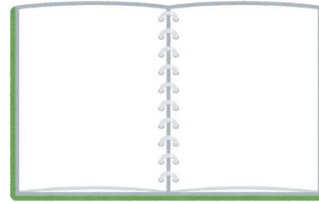
4-2 テーブル定義

データベースの構築



データベース生成

※ 最初、データベースは空



テーブル定義

(データ定義ともいう)

※ 最初、テーブルは空



データベース操作、問い合わせ



※ データベースシステムには、
たくさんのデータベースを作ることが
できる

データベースの運用の前に！

コンピュータ



リレーショナル
データベース
管理システム



記憶
装置

リレーショナル
データベース

あわせて

リレーショナルデータベースシステム

	id	name	teacher_name	student_name	score	created_at	updated_at
1	1	Database	K	KK	85	2014-09-13 00:46:17	{null}
2	2	Database	K	AA	75	2014-09-13 00:48:26	{null}
3	3	Database	K	LL	90	2014-09-13 00:48:26	{null}
4	4	Programming	A	KK	85	2014-09-13 00:48:26	{null}
5	5	Programming	A	LL	75	2014-09-13 00:48:26	{null}

	id	year	month	day	customer_name	product_id	qty	created_at
1	1001	2014	10	26	kaneko	1	10	2014-09-14 14:28:38
2	1002	2014	10	26	miyamoto	2	2	2014-09-14 14:28:38
3	1003	2014	10	27	kaneko	3	8	2014-09-14 14:28:38
4	1004	2014	10	27	kaneko	3	8	2014-09-14 14:28:38

データの種類ごとに分かれた、
たくさんの**テーブル**が格納される

テーブルを格納するには、**テーブル定義**が必要。

今日の授業で学ぶこと

Access 2013 の**デザインビュー**（Access だけで動くツール）
に頼らなくても

商品		
	フィールド名	データ型
🔑	ID	数値型
	商品名	短いテキスト
	単価	数値型

マウスと
キーボード

S Q L で**テーブル定義**ができるようになる（世界標準）

```
create table 商品 (  
    ID integer,  
    商品名 char,  
    単価 integer  
);
```

S Q L は
コマンド言語

4-2 テーブル定義とは

リレーショナルデータベースにおいて、

- ・ テーブル名
- ・ 各フィールドのフィールド名（属性名）
- ・ 各フィールドのデータ型

などを定義すること



ノートページ

確かに SQL は世界標準

■ SQLite3 システムの場合

```
1 create table 商品 (  
2     ID integer,  
3     商品 char,  
4     単価 integer  
5 );  
6
```

SQL によるテーブル定義

	ID	商品	単価
1	1	みかん	50
2	2	りんご	100
3	3	メロン	500

テーブルの中身の編集画面

■ Access 2013 システムの場合

クエリ1

```
create table 商品 (  
ID integer,  
商品名 char,  
単価 integer  
);
```

SQL によるテーブル定義 (同じ)

	ID	商品名	単価
	1	みかん	50
	2	りんご	100
	3	りんご	150
	4	メロン	500

テーブルの中身の編集画面

4-3 SQL によるテーブル定義

SQL のデータ型

ID	商品名	単価
1	みかん	50
2	りんご	100
3	りんご	150
4	メロン	500

フィールド フィールド フィールド

フィールドが3つ

リレーショナルデータベース
で、データ型というときは

各フィールドのデータの種類
のこと

SQL のデータ型は「英語のキーワード」

ID	商品名	単価
1	みかん	50
2	りんご	100
3	りんご	150
4	メロン	500

数値型 短いテキスト 数値型

← Access 2013 のデザイン
ビューを使うとき

integer char integer

← **SQL** を使うとき

4-3-1 SQL のデータ型



ノートページ

Access 2013 の主なデータ型	SQL のキーワード
短いテキスト	char
長いテキスト	text
数値型	integer, real
日付／時刻型	datetime
YesやNo	bit

※ 「**短いテキスト**」は
半角 255文字分までが目安

※ 整数は integer,
浮動小数点数は real

Access 2013 のデザインビューでのテーブル定義例

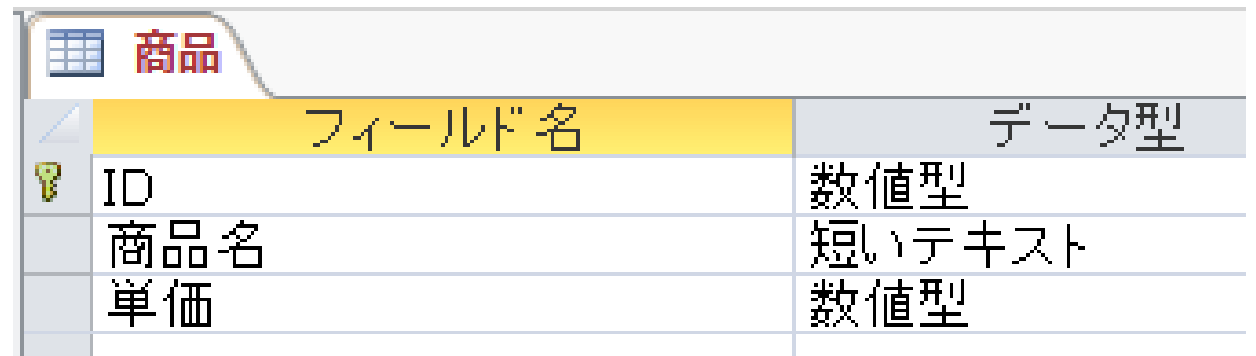
テーブル名：商品

ID	商品名	単価
1	みかん	50
2	りんご	100
3	りんご	150
4	メロン	500

テーブルの例

フィールド名	Access 2013 のデータ型
ID	数値型
商品名	短いテキスト
単価	数値型

Access 2013 のテーブル定義例



フィールド名	データ型
ID	数値型
商品名	短いテキスト
単価	数値型

Access 2013 のデザインビューでのテーブル定義例

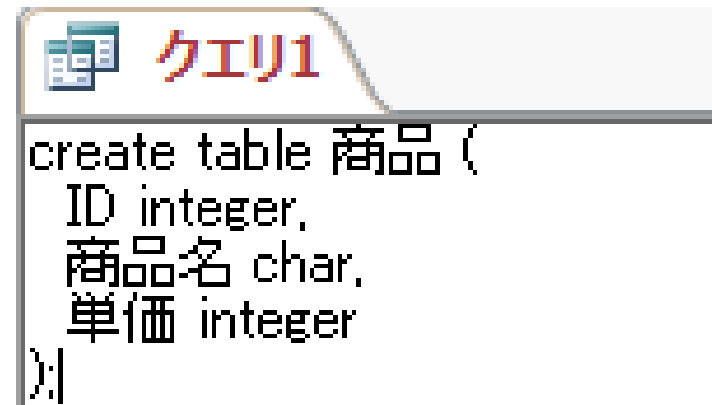
テーブル名：商品

ID	商品名	単価
1	みかん	50
2	りんご	100
3	りんご	150
4	メロン	500

テーブルの例

```
create table 商品 (  
  ID integer,  
  商品名 char,  
  単価 integer  
);
```

SQL でのテーブル定義例



Access 2013 の画面例（同じ）

テーブル定義例を比べてみると

```
create table 商品 (  
  ID integer,  
  商品名 char,  
  単価 integer  
);
```

SQL でのテーブル定義例

商品		
	フィールド名	データ型
🔑	ID	数値型
	商品名	短いテキスト
	単価	数値型

Access 2013 のデザインビュー画面例

テーブル定義例を比べてみると

テーブル名

```
create table 商品 (  
  ID integer,  
  商品名 char,  
  単価 integer  
);
```

フィールド名とデータ型

SQL でのテーブル定義例

テーブル名

商品		
	フィールド名	データ型
🔑	ID	数値型
	商品名	短いテキスト
	単価	数値型

フィールド名とデータ型

Access 2013 のデザインビュー画面例

4-3-2 SQL によるテーブル定義の例

テーブル定義の例

```
create table 商品 (  
  ID integer,  
  商品名 char,  
  単価 integer  
);
```

```
create table 購入 (  
  ID integer,  
  名前 char,  
  商品番号 integer  
);
```



SQLは、リレーショナルデータベース管理システムの
ほぼ全機能を扱えるデータベース言語！

機能	SQLコマンドの例
テーブル定義	CREATE TABLE 今日はここを説明
参照整合性制約	CREATE TABLE 文の中のFOREIGN KEY 句, REFERENCES 句
索引の作成	CREATE INDEX
新規データの登録	INSERT INTO, CREATE TABLE AS
問い合わせ（クエリ）	SELECT FROM WHERE
更新（挿入、削除、変更）	INSERT INTO, DELETE FROM WHERE, UPDATE SET WHERE

4-4 リレーショナルデータベースの テーブル

リレーショナルデータベースのテーブルは

テーブル名：商品

ID	商品名	単価
1	みかん	50
2	りんご	100
3	りんご	150
4	メロン	500

テーブルの例

リレーショナルデータベースのテーブルは

テーブル名：福山駅行き

時	分
8	0
8	20
8	45
12	30
17	20
17	40

- ☑ リレーショナルデータベースでは、1つのセルに1つの値（鉄則）

リレーショナルデータベースで扱えないテーブルの例

時	分
8	0, 20, 45
12	30
17	20, 40

- ☐ 1つのセルに複数の値を入れることはない

リレーショナルデータベースのテーブルは（念押し）

テーブル名：福山駅行き

時	分
8	0
8	20
8	45
12	30
17	20
17	40

- ☒ リレーショナルデータベースでは、レコード（行）を単位とする（鉄則）

リレーショナルデータベースで扱えないテーブルの例

時	分
8	0
	20
	45
12	30
17	20
	40

- ☐ マルチカラムにはしない

リレーショナルデータベースのテーブルは

テーブル名：福山駅行き

時	分
8	0
8	20
8	45
12	30
17	20
17	40

だからこそ

- ◆ データ型はシンプル
配列、構造体、共用体、
リスト、ポインタなどが無い
- ◆ SQL もシンプル

- ☑ リレーショナルデータベースでは、
1つのセルに1つの値（鉄則）

テーブル定義に関する研究課題

- ◆ ベストなテーブル定義は何か？
→ 難問 未解決
- ◆ テーブル定義では何に気を付けたらよいか？
今わかっていること
同じ情報を、複数記録するようなテーブル定義は良くない
⇒ 今の技術で、ある程度解決可能
- ◆ データがたくさん集まったとき、区別、分類、選別、整理しないといけない（そうしないと、テーブルが作れない）
⇒ 今の技術で、ある程度解決可能

4-5 問い合わせ（クエリ）の紹介

SQLの使い方



ID	商品名	単価
1	みかん	50
2	りんご	100
3	りんご	150
4	メロン	500



データベースシステムに
SQLコマンドを**送る**



結果が返ってくる

データベース
利用者

SQLで、テーブルをそのまま表示

ID	商品名	単価
1	みかん	50
2	りんご	100
3	りんご	150
4	メロン	500

元のテーブルの
まま表示

1	<code>select * from 商品;</code>
2	
3	
4	

クエリ1		
ID	商品名	単価
1	みかん	50
2	りんご	100
3	りんご	150
4	メロン	500



問い合わせ
(クエリ)



結果

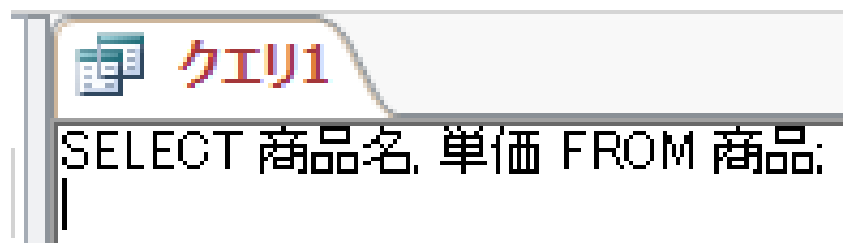
フィールドの表示／非表示（射影）の例

ID	商品名	単価
1	みかん	50
2	りんご	100
3	りんご	150
4	メロン	500

フィールドの表示／非表示 （「射影」といいます）



問い合わせ
(クエリ)



結果

商品名	単価
みかん	50
りんご	100
りんご	150
メロン	500

レコードの絞り込み（選択）の例

ID	商品名	単価
1	みかん	50
2	りんご	100
3	りんご	150
4	メロン	500

全体で「射影＋選択」

レコードの絞り込み
(選択)



```
1 select 商品名, 単価 from 商品
2 where 単価 > 80;
3
```

商品名	単価
りんご	100
りんご	150
メロン	500

問い合わせ
(クエリ)

結果

※ 途中で改行してもいいし、改行しなくてもよい (SQL のルール)

SQL を用いた問い合わせ（クエリ）

- 「**問い合わせ（クエリ）**」とは、データベースの**検索（抽出）**、**分類**、**集計・集約**を行うこと
- **問い合わせ（クエリ）**の結果は、**必ず1つ**のテーブルになる
- キーワード: **select, from, where**

複数のテーブルを1つにまとめる（結合）例

商品

ID	商品名	単価
1	みかん	50
2	りんご	100
3	りんご	150
4	メロン	500

購入

ID	名前	商品
1	X	3
2	Y	1

SQL でテーブル名を2つ書く

```
select * from 商品, 購入;
```



1	みかん	50	1	X	3
1	みかん	50	2	Y	1
2	りんご	100	1	X	3
2	りんご	100	2	Y	1
3	りんご	150	1	X	3
3	りんご	150	2	Y	1
4	りんご	500	1	X	3
4	りんご	500	2	Y	1