

データベース

自治体での住民登録、図書館での蔵書管理など、大量のデータが利用しやすく管理されている。このように、ある目的のためにデータを蓄積して利用価値を高めたものをデータベースという。

データバールの変遷

①カード型

ページ1枚に1件のデータを格納する構造のデータベース

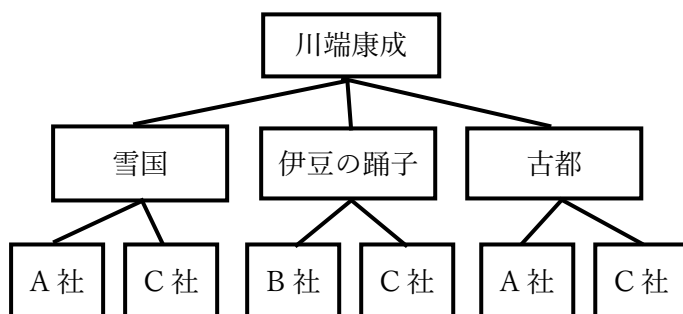
ID	x x x x x x x x
書籍名	雪国
著者名	川端康成
出版社	A 社、C 社

ID	x x x x x x x x
書籍名	伊豆の踊子
著者名	川端康成
出版社	B 社、C 社

ID	x x x x x x x x
書籍名	古都
著者名	川端康成
出版社	A 社、C 社

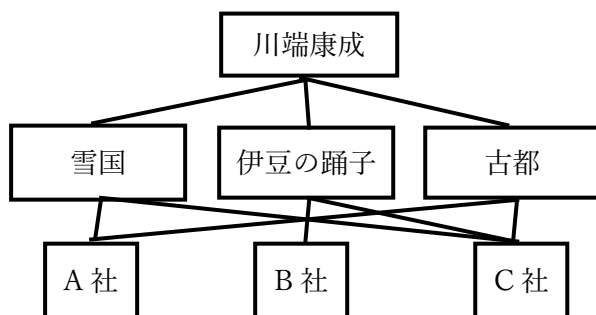
②階層型(ツリー型)

親から子へツリーの形になるような構造のデータベース。データに重複が生じる。



③ネットワーク型

重複が生じないようにリンクをさせる構造のデータベース。更新作業が大変になる。



④リレーショナル型

外部キーを使って、単純な構造の表と表をつなげ合わせるデータベース。これが現在の主流のデータベースの型である。

書籍表

著者名	書籍名	出版社コード
川端康成	雪国	001
川端康成	雪国	003
川端康成	伊豆の踊子	002
川端康成	伊豆の踊子	003
川端康成	古都	001
川端康成	古都	003

出版社表

出版社コード	出版社
001	A 社
002	B 社
003	C 社

●その他のデータベース

SNS などの膨大な数のデータを扱う場合、リレーショナルデータベースでは対応できない場合があり、リレーショナル以外のデータベースの総称を NoSQL と呼んでいる。NoSQL では、キーバリュー型、カラム指向型、グラフ指向型などのデータモデルがある。

(参考) <https://www.kagoya.jp/howto/it-glossary/server/nosql/>

⑤キーバリュー型

データを識別する一意の「キー」と「キー」と組になる「バリュー」だけで構成されるデータベースの型である。

キー	バリュー
12345678	(書籍名:雪国、著者名:川端康成、出版社:A 社、版:2)
23456781	(書籍名:雪国、著者名:川端康成、出版社:C 社)
34567812	(書籍名:伊豆の踊子、著者名:川端康成、出版社:B 社)
45678123	(書籍名:伊豆の踊子、著者名:川端康成、出版社:C 社)

⑥カラム指向型

キーバリュー型にカラム(列)が追加されたデータベースの型である。任意の列をまとめて処理できる。

キー	書籍名	著者名	出版社
12345678	雪国	川端康成	A 社
23456781	雪国	川端康成	C 社
34567812	伊豆の踊子、	川端康成	B 社
45678123	伊豆の踊子	川端康成:	C 社

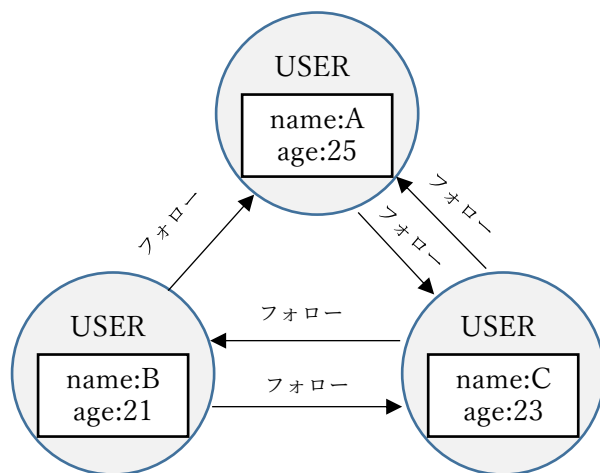
⑦ドキュメント型

JSON や XML などのドキュメント形式で管理するデータベースの型である

キー	ドキュメント
12345678	{shosekimei:'雪国' choshamei:'川端康成' shuppansha:'A 社' }
23456781	
34567812	
45678123	

⑧グラフ型

グラフ構造を備えたデータベースの型である。データの構造が従来のリレーショナルではなくネットワーク状になっている場合に、格納・検索の面で威力を発揮する。例えば、友達の友達を探すときに、100万人規模の会員の SNS の中から友達を探して行くとリレーショナル型では時間がかかる。グラフ型では隣接しているノードを表示するため時間がかからない。



参考 グラフデータベースとは何か <https://www.imagazine.co.jp/12805-2/>

データベースの環境構築

データベースの環境を構築する方法を3つ紹介する

① Windows マシンに SQL を組み込み、コマンドプロンプトから起動する。

ここでは2つのオープンソースデータベースを紹介する

●SQLite

コンパクトでサーバレスの DBMS。SQLite の公式ページ(<https://www.sqlite.org/>)の Download からコマンドラインツールをダウンロードして、プログラムを配置する。

詳しくは https://www.javadrive.jp/sqlite/#section_install に書かれている

●MySQL

高速で信頼性が高く、習得が容易でクライアントサーバーモデルの DBMS。MySQL のサイト <https://dev.mysql.com/downloads/windows/installer/5.7.html> からインストーラーをダウンロードして、自分の PC にインストールする。

詳しくは、<https://prog-8.com/docs/mysql-env-win> に書かれている。

② Paiza IO で MySQL を使う。

paiza.io の MySQL のページ <https://paiza.io/ja/projects/new?language=mysql> にアクセスする

③ Google Colab で SQL をインポートして、マジックコード%%sql を入力して使う。

●ipython-sql というライブラリをインストールして、notebook のマジックコマンドで SQL が使えるライブラリを用いる

詳しくは、<https://qiita.com/jinta/items/fb79f9f9b65b241e92b2> に書かれている。

※Pandas と組み合わせるときれいに出力できる。

●%reload_ext sql

%sql sqlite:///memory:

と入力して実行し、SQLite のメモリデータベースを作成して、%%sql のマジックコードを使う

詳しくは、<https://zenn.dev/uinoue/articles/750087c8352c52> に書かれている。

データベース演習

下の2つの表(書籍表、分類表)を作って、結合、選択、射影の操作をする。

書籍表

書籍 ID	書籍名	著者名	分類コード
123456	伊豆の踊子	川端康成	910
234567	斜陽	太宰治	910
345678	レ・ミゼラブル	ヴィクトル・ユーゴー	950
456789	ロミオとジュリエット	ウィリアム・シェイクスピア	930

分類表

分類コード	分類
910	日本文学
930	英米文学
950	フランス文学

結合（分類コードをキーにして、書籍表に分類表を結合する）

書籍 ID	書籍名	著者名	分類コード	分類コード	分類
123456	伊豆の踊子	川端康成	910	910	日本文学
234567	斜陽	太宰治	910	910	日本文学
345678	レ・ミゼラブル	ヴィクトル・ユーゴー	950	950	フランス文学
456789	ロミオとジュリエット	ウィリアム・シェイクスピア	930	930	英米文学

選択（書籍表から、分類コード910を条件にして表示する）

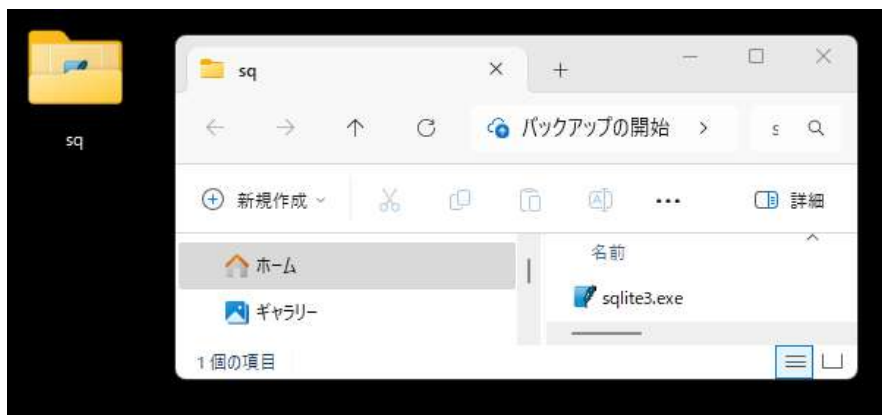
書籍 ID	書籍名	著者名	分類コード
123456	伊豆の踊子	川端康成	910
234567	斜陽	太宰治	910

射影（書籍表から、書籍名、著者名）だけを表示させる

書籍名	著者名
伊豆の踊子	川端康成
斜陽	太宰治

① SQLite で操作する

用意した sqlite3.exe ファイルが入った sq フォルダをデスクトップにコピーする。



スタートボタンを右クリックして、ターミナルやコマンドプロンプトなどを実行する
コマンドラインに「cd 」と入力して

```
PowerShell 7.4.6  
PS C:\Users\[redacted]> cd C:\Users\[redacted]\Desktop\sq
```

デスクトップのsqフォルダをウィンドウにドラッグすると、フォルダのパスが入力される

```
PowerShell 7.4.6  
PS C:\Users\[redacted]> cd C:\Users\[redacted]\Desktop\sq
```

データベースを作成する

```
PS C:\Users\[redacted]\Desktop\sq> sqlite3 db1.db  
SQLite version 3.29.0 2019-07-10 17:32:03  
Enter ".help" for usage hints.  
sqlite>
```

一度抜ける

```
sqlite> .exit
```

フォルダにデータベースが作成される



データベースファイルを読み込んで sqlite を実行する

```
PS C:\Users\[redacted]\Desktop\sq> sqlite3 db1.db
```

テーブルの作成

●書籍表を作る

カラム名、カラムの型を定義して、shosekihyou テーブルを作る

```
sqlite> create table shosekihyou(id integer,title text,chosha text,bunruicode integer);
```

※ここでは文字列を text 型にしているが短い文字列の場合 varchar 型にした方効率が良い

shosekihyou テーブルにレコードを追加する

```
sqlite> insert into shosekihyou values(123456,'伊豆の踊子','川端康成',910);
sqlite> insert into shosekihyou values(234567,'斜陽','太宰治',910);
sqlite> insert into shosekihyou values(345678,'レ・ミゼラブル','ヴィクトル・ユーゴー',950);
sqlite> insert into shosekihyou values(456789,'ロミオとジュリエット','ウィリアム・シェイクスピア',930);
```

カラム名が表示されるように設定を変える

```
sqlite> .headers on
```

shosekihyou テーブルの全カラムを表示する

```
sqlite> select * from shosekihyou;
id|title|chosha|bunruicode
123456|伊豆の踊子|川端康成|910
234567|斜陽|太宰治|910
345678|レ・ミゼラブル|ヴィクトル・ユーゴー|950
456789|ロミオとジュリエット|ウィリアム・シェイクスピア|930
```

●分類表を作る

カラム名、カラムの型を定義して、bunruihyou テーブルを作る

```
sqlite> create table bunruihyou(buinruicode integer,bunrui text);
```

レコードを追加する

```
sqlite> insert into bunruihyou values(910,'日本文学');
sqlite> insert into bunruihyou values(930,'英米文学');
sqlite> insert into bunruihyou values(950,'フランス文学');
```

bunruihyou テーブルの全カラムを表示する

```
sqlite> select * from bunruihyou;
910|日本文学
930|英米文学
950|フランス文学
```

●結合

2つの表を bunruicode をキーにして結合する

<コード>

```
select * from shosekihyou inner join bunruihyou on shosekihyou.bunruicode = bunruihyou.bunruicode;
```

<操作画面>

```
sqlite> select * from shosekihyou inner join bunruihyou on shosekihyou.bunruicode = bunruihyou.bunruicode;
```

<実行結果>

```
id|title|chosha|bunruicode|bunruicode|bunrui  
123456|伊豆の踊子|川端康成|910|910|日本文学  
234567|斜陽|太宰治|910|910|日本文学  
345678|レ・ミゼラブル|ヴィクトル・ユーゴー|950|950|フランス文学  
456789|ロミオとジュリエット|ウィリアム・シェイクスピア|930|930|英米文学
```

●選択

shosekihyou テーブルの bunruicode が910のレコードを表示する

<コード>

```
select * from shosekihyou where shosekihyou.bunruicode=910;
```

```
sqlite> select * from shosekihyou where shosekihyou.bunruicode=910;
```

実行結果

```
id|title|chosha|bunruicode  
123456|伊豆の踊子|川端康成|910  
234567|斜陽|太宰治|910
```

※結合した後に選択をした場合、bunruicode のカラム名が2つになるため、shosekihyou.bunruicode=910 のようにテーブルの情報を追加する必要がある。

■射影

shosekihyou テーブルの title と chosha のカラムを表示する

<コード>

```
select title,chosha from shosekihyou;
```

<操作画面>

```
sqlite> select title,chosha from shosekihyou;
```

<実行結果>

```
title|chosha  
伊豆の踊子|川端康成  
斜陽|太宰治  
レ・ミゼラブル|ヴィクトル・ユーゴー  
ロミオとジュリエット|ウィリアム・シェイクスピア
```


② Paiza.io のMySQL で操作をする

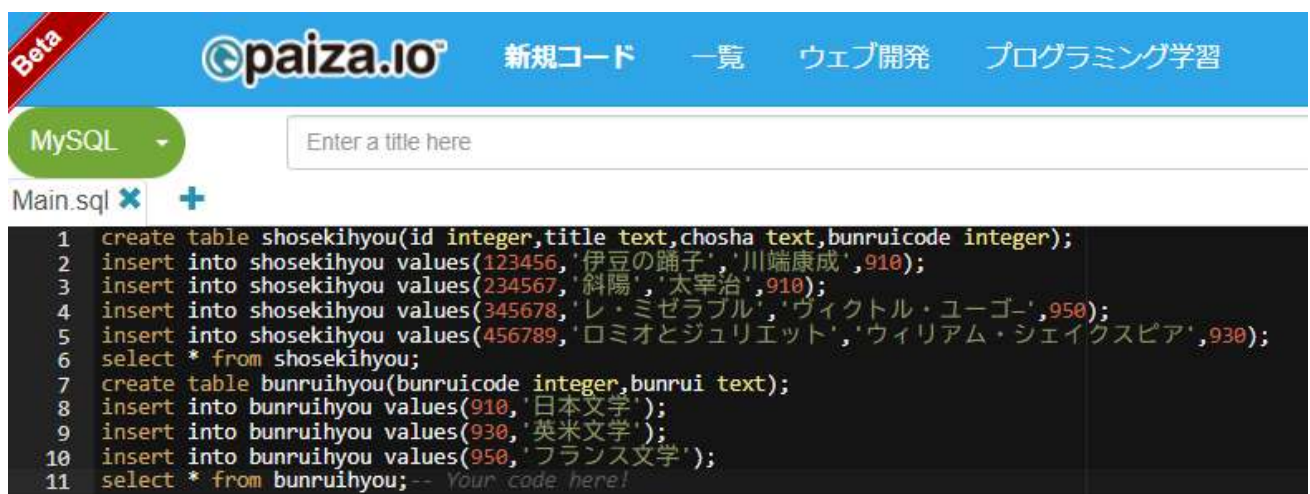
テーブルの作成

Main.sql に shosekihyou テーブルと bunruihyou テーブルを作成して、全カラムを表示する

<コード>

```
create table shosekihyou(id integer,title text,chosha text,bunruicode integer);
insert into shosekihyou values(123456,'伊豆の踊子','川端康成',910);
insert into shosekihyou values(234567,'斜陽','太宰治',910);
insert into shosekihyou values(345678,'レ・ミゼラブル','ヴィクトル・ユーゴー',950);
insert into shosekihyou values(456789,'ロミオとジュリエット','ウィリアム・シェイクスピア',930);
select * from shosekihyou;
create table bunruihyou(bunruicode integer,bunrui text);
insert into bunruihyou values(910,'日本文学');
insert into bunruihyou values(930,'英米文学');
insert into bunruihyou values(950,'フランス文学');
select * from bunruihyou;
```

<操作画面>



<実行結果>

id	title	chosha	bunruicode
123456	伊豆の踊子	川端康成	910
234567	斜陽	太宰治	910
345678	レ・ミゼラブル	ヴィクトル・ユーゴー	950
456789	ロミオとジュリエット	ウィリアム・シェイクスピア	930
bunruicode	bunrui		
910	日本文学		
930	英米文学		
950	フランス文学		

■結合

分類コードをキーにして2つの表を結合をする

<コード>

```
create table shosekihyou(id integer,title text,chosha text,bunruicode integer);
insert into shosekihyou values(123456,'伊豆の踊子','川端康成',910);
insert into shosekihyou values(234567,'斜陽','太宰治',910);
insert into shosekihyou values(345678,'レ・ミゼラブル','ヴィクトル・ユーゴー',950);
insert into shosekihyou values(456789,'ロミオとジュリエット','ウィリアム・シェイクスピア',930);
create table bunruihyou(bunruicode integer,bunrui text);
insert into bunruihyou values(910,'日本文学');
insert into bunruihyou values(930,'英米文学');
insert into bunruihyou values(950,'フランス文学');
select * from shosekihyou inner join bunruihyou on shosekihyou.bunruicode = bunruihyou.bunruicode;
```

<操作画面>

```
1 create table shosekihyou(id integer,title text,chosha text,bunruicode integer);
2 insert into shosekihyou values(123456,'伊豆の踊子','川端康成',910);
3 insert into shosekihyou values(234567,'斜陽','太宰治',910);
4 insert into shosekihyou values(345678,'レ・ミゼラブル','ヴィクトル・ユーゴー',950);
5 insert into shosekihyou values(456789,'ロミオとジュリエット','ウィリアム・シェイクスピア',930);
6 create table bunruihyou(bunruicode integer,bunrui text);
7 insert into bunruihyou values(910,'日本文学');
8 insert into bunruihyou values(930,'英米文学');
9 insert into bunruihyou values(950,'フランス文学');
10 select * from shosekihyou inner join bunruihyou on shosekihyou.bunruicode = bunruihyou.bunruicode;
```

<実行結果>

id	title	chosha	bunruicode	bunruicode	bunrui
123456	伊豆の踊子	川端康成	910	910	日本文学
234567	斜陽	太宰治	910	910	日本文学
345678	レ・ミゼラブル	ヴィクトル・ユーゴー	950	950	フランス文学
456789	ロミオとジュリエット	ウィリアム・シェイクスピア	930	930	英米文学

■選択

shosekihyou テーブルの分類コード 910 のデータを表示する

<コード>

```
create table shosekihyou(id integer,title text,chosha text,bunruicode integer);
insert into shosekihyou values(123456,'伊豆の踊子','川端康成',910);
insert into shosekihyou values(234567,'斜陽','太宰治',910);
insert into shosekihyou values(345678,'レ・ミゼラブル','ヴィクトル・ユーゴー',950);
insert into shosekihyou values(456789,'ロミオとジュリエット','ウィリアム・シェイクスピア',930);
select * from shosekihyou where bunruicode=910;
```

<操作画面>

```
1 create table shosekihyou(id integer,title text,chosha text,bunruicode integer);
2 insert into shosekihyou values(123456,'伊豆の踊子','川端康成',910);
3 insert into shosekihyou values(234567,'斜陽','太宰治',910);
4 insert into shosekihyou values(345678,'レ・ミゼラブル','ヴィクトル・ユーゴー',950);
5 insert into shosekihyou values(456789,'ロミオとジュリエット','ウィリアム・シェイクスピア',930);
6 select * from shosekihyou where bunruicode=910;
```

<実行結果>

id	title	chosha	bunruicode
123456	伊豆の踊子	川端康成	910
234567	斜陽	太宰治	910

■射影

shosekihyou テーブルの書籍名と著者名だけ表示する

<コード>

```
create table shosekihyou(id integer,title text,chosha text,bunruicode integer);
insert into shosekihyou values(123456,'伊豆の踊子','川端康成',910);
insert into shosekihyou values(234567,'斜陽','太宰治',910);
insert into shosekihyou values(345678,'レ・ミゼラブル','ヴィクトル・ユーゴー',950);
insert into shosekihyou values(456789,'ロミオとジュリエット','ウィリアム・シェイクスピア',930);
select title,chosha from shosekihyou;
```

<操作画面>

```
1 create table shosekihyou(id integer,title text,chosha text,bunruicode integer);
2 insert into shosekihyou values(123456,'伊豆の踊子','川端康成',910);
3 insert into shosekihyou values(234567,'斜陽','太宰治',910);
4 insert into shosekihyou values(345678,'レ・ミゼラブル','ヴィクトル・ユーゴー',950);
5 insert into shosekihyou values(456789,'ロミオとジュリエット','ウィリアム・シェイクスピア',930);
6 select title,chosha from shosekihyou;
```

<実行結果>

title	chosha
伊豆の踊子	川端康成
斜陽	太宰治
レ・ミゼラブル	ヴィクトル・ユーゴー
ロミオとジュリエット	ウィリアム・シェイクスピア

【課題】結合した表を使って、選択、射影をこなさい。