

## データベース

### shop データベース

id	INTEGER	ID 管理 (1 対 1 対応の重複無し)	AUTO_INCREMENT NOT NULL PRIMARY KEY
name	VARCHAR(20)	商品名 (20 文字以下)	
price	INTEGER	価格 (単位: 円)	
genre	INTEGER	ジャンル(数字と 1 対 1 対応)	
stock	INTEGER	在庫数 (単位: 個) 含 reservation	
reservation	INTEGER	予約数 (単位: 個)	

### reservation データベース

id	INTEGER	ID 管理 (1 対 1 対応の重複無し)	AUTO_INCREMENT NOT NULL PRIMARY KEY
grade	INTEGER	学年 ( $1 \leq \text{grade} \leq 5$ )	
department	INTEGER	ME(0), IE(1), CA(2), MC(3), IC(4), EC(5)	
number	INTEGER	出席番号	
name	VARCHAR(30)	名前	
purchase	VARCHAR(100)		
price	INTEGER	総計 (単位: 円)	
password	VARCHAR(100)	パスワード (encrypt 済)	

## メソッド

### <データベース関連>

#### ◎ setDataShop

引数：name, price, genre, stock, reservation

処理：shop.db にデータ登録

#### ◎ setDataReservation

引数：grade, department, number, name, purchase, price, password

処理：reservation.db にデータ登録

#### ◎ deleteData

引数：database, id

処理：database データベースのレコードの id カラムが id であるデータを削除

#### ◎ getDBInfo

引数：database, column, id=None

処理：database データベースの id カラムが id である column カラムのデータを取得

→ id を省略することで、レコードを指定せず取得

→ column を\*にすることで、column を指定せず取得

戻り値：タプルのリスト（注意！）

### <パスワード関連>

#### ◎ encryptPassword

引数：original\_password

処理：暗号化したいメッセージを引数に渡すと、暗号化したパスワードを返す。

→ repr(戻り値) をデータベースに追加しないとエラー

#### ◎ decryptPassword

引数：cipher\_data

処理：暗号化されたメッセージを引数に渡すと、暗号化したパスワードを返す。

→ eval(“データベースから取得した値”) を引数に渡さないとエラー