データベース

1章SQLとデータベース

データベース

データベース（DB）…データをコンピューターを使って効率よくアクセスできるようにしたデータの集まり。

データベースマネージメント（DBMS）・・・データベースを管理するコンピュータシステム。

DBMSの場合

|  |  |
| --- | --- |
| 多人数でデータを共有するのに向かない | 大量データを扱える形式になってない |
| 読み書きを自動化するのにプログラミングが必要 | 万が一の事故に対応できない |

DBMSの例

　　・Oracle Database・・・Oracle社のRDBMS

　　・SQL Server・・・Microsoft社のRDBMS

　　・DB2・・・IBM社のRDBMS

　　・PostgreSQL・・・オープンソースのRDBMS

　　・MySQL・・・オープンソースのRDBMS（2010年からOracle社が開発）

RDBMS・・・リレーショナルデータベースマネージメント、関係データベースとも呼ばれる現在もっとも使われているDBMS。列と行からなる２次元表の形式（表、テーブルと呼ぶ）でデータを管理する。縦は列、横は行（レコードともいう）と呼び、必ず行単位でデータを読み書きする。

システムの構成はクライアント/サーバー型（C/S型）

※１つのセルの中には１つのデータしか入れられない。

RDMSの例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 商品ID | 商品名 | 商品分類 | 販売単価 | 仕入単価 | 登録日 |
| 0001 | Tシャツ | 衣類 | 1000 | 500 | 2009-09-20 |
| 0002 | 穴あけパンチ | 事務用品 | 320 | 320 | 2009-09-11 |
| 0003 | カッターシャツ | 衣類 | 4000 | 2800 |  |
| 0004 | 包丁 | キッチン用品 | 2800 | 2000 | 2009-09-20 |
| 0005 | 圧力鍋 | キッチン用品 | 6800 | 5000 | 2009-01-15 |

|  |
| --- |
| 商品ID |
| 0001 |
| 0002 |
| 0003 |

これは縦なので列（カラム）、商品IDの列と呼ぶ。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0001 | Tシャツ | 衣類 | 1000 | 500 | 2009-09-20 |

これが行（レコード）、商品ID「0001」のレコードと呼ぶ。

SQL

SQL・・・RDBを操作する言語。検索や登録、削除などのデータベースの作業ができる。

　　　　　ISO（国家標準化機構）が定めた標準規格があり、それに準じたものを標準SQLと呼ぶ。

SQL文・・・いくつかのキーワードとテーブル名や列名などを組み合わせた１つの文で特別な分、

　「テーブルを検索する」、「このテーブルを参照する」など様々な意味を持つものがある

DDL・・・データ定義言語、データを格納する入れ物、データベースを作成したり削除したりできる。

・データベースの作成　・データベースの削除　・データベースの構成の変更

DML・・・データ操作言語、テーブルの行を検索したり変更したりできる。

・テーブルから行を検索　・テーブルの新規行を登録　・テーブルの更新、削除

DCL・・・データ制御言語、データベースの変更を確定したり取り消ししたりするもの。

・データベースに対しておこなった変更の確定、削除　・ユーザーに権限を与える、奪う。

SQLを使うときのルール

　・文の最後に　；　をつける

　・大文字・小文字は区別されない（基本は大文字）

　 　・単語は半角スペースか開業で区切る

　・定数の書き方には決まりがある

※定数・・・SQL文の中にじかに書く文字列や数値

テーブルの作成

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 商品ID | 商品名 | 商品分類 | 販売単価 | 仕入単価 | 登録日 |
| 0001 | Tシャツ | 衣類 | 1000 | 500 | 2009-09-20 |
| 0002 | 穴あけパンチ | 事務用品 | 320 | 320 | 2009-09-11 |
| 0003 | カッターシャツ | 衣類 | 4000 | 2800 |  |
| 0004 | 包丁 | キッチン用品 | 3000 | 2000 | 2009-09-20 |
| 0005 | 圧力鍋 | キッチン用品 | 6800 | 5000 | 2009-01-15 |
| 0006 | フォーク | キッチン用品 | 500 |  | 2009-09-20 |
| 0007 | おろしがね | キッチン用品 | 880 | 790 | 2008-04-28 |
| 0008 | ボールペン | 事務用品 | 100 |  | 2009-11-11 |

データ型の指定

・INTEGER型・・・整数を入れる列に指定する型

・CHAR型・・・文字列を入れる列に指定する型（固定数文字列）

※使うときに「CHAR（数字）」この数字の所に文字数を入れる。文字数が数字より小さい 　　と空白が格納される。

・VARCHAR型・・・文字列を入れる列に指定する型（可変長文字列）

・DATE型・・・火付けを入れる列に指定する型

※固定数文字列と可変長文字列の違い・・・可変長文字列は上の数字より小さい文字でも空白が格納されない

データベースの作成

CLEATE　DATABASE文

　　・CREATE　DATABASE　＜データベース名＞；・・・データベース名のタイトル設定

CREATE　DATABASE　‘COOKIE’

　　・CREATE　TABLE　＜テーブル名＞

・・・テーブル名の設定

※テータベースやテーブル、列などで使える文字は「半角のアルファベット」、「半角数字」、「アンダーバー」のみ。

・DROP TABLE＜テーブル名＞

・・・作ったテーブルの削除コマンド

　　・ALTER　TABLE＜テーブル名＞　ADD　COLMUN＜列の定義＞

・・・テーブルの中に新しい列を作成するコマンド

　 　　・ALTER　TABLE＜テーブル名＞　DROP　COLMUN＜列の定義＞

・・・既存の列の削除コマンド

・PRIMARY KEY

・・・プライマリーキーと呼ばれ、CREATE TABLEの最後につける。

・START　TRANSACTION

・・・行追加の開始する命令コマンド

　　・RENAME　TABLE

・・・テーブル名の変更コマンド

例

　　CREATE　DATABASE　shop；←データベース名

　CREATE　TABLE shohin←テーブル名

(shohin\_id 　　　　CHAR(4)←4文字のID 　NOT NULL,←空白はいらないよって意味

　　　 shohin\_mei VARCHAR(100) 　　　　 NOT NULL,

shohin\_bunrui VARCHAR(32) 　　　　NOT NULL,

hanbai\_tanka INTEGER 　　 ,←数字なので空白つかないのでNOT　NULLいらない

shire\_tanka INTEGER ,

torokubi DATE ,

PRIMARY KEY (shohin\_id));←主キー

↓結果

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Shohin\_id | Shohim\_mei | Shohin\_bunrui | Hanbai\_tanka | Shire\_tanka | torokubi |

となる。

テーブルへのデータ登録時

START TRANSACTION;

INSERT INTO Shohin VALUES ('0001', 'Tシャツ' ,'衣服', 1000, 500, '2009-09-20');

INSERT INTO Shohin VALUES ('0002', '穴あけパンチ', '事務用品', 500, 320, '2009-09-11');

これでshohinテーブルに2つ追加された。