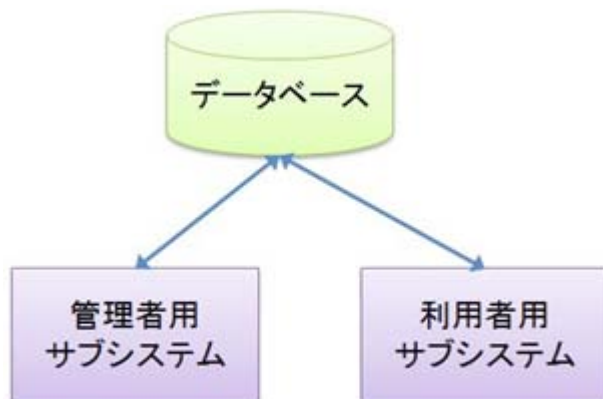


第6回で提示したモデリング課題で、情報システムを次の図のように実装することになったとします。



管理用サブシステムと利用者用サブシステムがデータベースにアクセスして処理を行います。コミュニティホールの職員は管理用サブシステムを使用し、一般市民とIDを取得した市民は利用者用サブシステムを使用します。このようなシステム構成において、情報システムが扱う対象(ステレオタイプのエンティティ)は、データベースに保持されることになります。

そこで、詳細設計に入る前に、データベースについて学習しておきます。詳細設計を行うためには、実装についての知識が必要だからです。ただし、詳しくは「データベース論」という科目がありますから、そちらで学習してください。ここでは以後の実習を進めるために必要な知識のみを学習します。

学習目標

今回の学習モジュールを修了すると、次のようなことができるようになるはずです。そうなるように学習しましょう。

- 関係データベースがどのようなデータベースか、その操作言語SQLはどのような特徴を持つか説明できる
- 関係データベースにおけるテーブルがどのようなものか説明できる
- SQLのコマンドで、テーブルの作成と確認、削除ができる
- SQLのコマンドで、テーブルにデータを挿入できる
- SQLのコマンドで、データ検索ができる

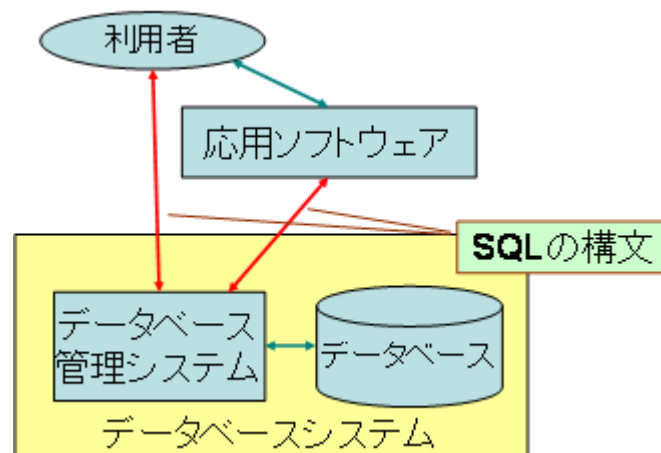
ここでは、MySQLというデータベースを使用して実習を行います。学習目標で示されているSQLのコマンドはMySQLでも他のSQLでも同じです。

関係データベースとSQL

データベースは複数のアプリケーションソフトやユーザによって共有されるデータの集まりで、その管理システムまでを含めて「データベース」と呼ぶこともあります。表計算ソフトでもデータの集まりを管理することができますが、データベースは通常はサーバとして機能し、複数のアプリケーションやユーザから並行してアクセスされてもきちんと処理ができ、障害回復の機能をもっている点が大きく異なります。

データベースの主流は「関係モデル」です。これは1970年にE.F.Codによって提案されたもので、「リレーション」と呼ぶ表（テーブル）の形でデータを管理するものです。関係モデルによるデータベースを「リレーショナルデータベース」あるいは「関係データベース」と呼びます。他には、データを木構造で整理する「階層モデル」、データ同士を結んで管理する「ネットワークモデル」などがあります。

関係データベースを操作する言語としてSQL(Structured Query Language)があります。日本語では「構造化問い合わせ言語」といった意味になります。SQLは関係データベースを操作するための標準言語となっており、ISO（国際標準規格）やJIS（日本工業規格）でも定義されています。SQLを使ってデータベースの作成や削除、データの挿入、データの検索、データの削除、データの更新などを行うことができます。データベースと応用ソフトウェア(アプリケーションソフトウェア)の関係を以下の図に示します。



せまい意味でのデータベースはデータの集まりの部分です。それを管理するシステムがデータベース管理システムであり、データベースとデータベース管理システムをまとめて「データベースシステム」と呼びます。ただし、データベースシステムのことを単にデータベースと呼ぶこともありますから注意してください。データベースを操作するためにはデータベース管理システムにSQLで記述したコマンドを入力します。これはSQLモニタという画面を使って利用者が直接SQLのコマンドを入力する場合と、応用ソフトウェアからSQLのコマンドを送信する場合があります。いずれにしてもSQLの構文を使うことになりますから、SQLを知っておくとよいでしょう。

この授業ではSQLによるデータベースの1つであるMySQLを使って実習を行います。また、Java言語からSQLのコマンドを送信する方法で応用ソフトウェアを作成します。

データを格納するテーブル

1つのデータベースは複数のテーブルから成ります。たとえば、学生がどの科目を履修しているかという情報を扱うデータベースは、以下のように「学生」「科目」「履修登録」といった3つのテーブルで管理することができます。

学生(Student)

学籍番号	姓	名
09R001	帝京	太郎
09R002	理工	次郎
08R001	通信	三郎

科目(Subject)

科目ID	科目名	担当教員
k001	情報科学演習1	わたなべ
k002	情報科学演習2	ささき
k003	情報科学演習3	ささき
k004	情報科学演習4	わたなべ

履修登録(Registration)

科目ID	学籍番号
k001	09R001
k001	09R002
k002	09R001
k003	08R001
k004	08R001

Coddが提案した関係データベースでは、テーブルのことを「関係(リレーション)」と呼びます。また、学生というテーブルにある「学籍番号」「姓」「名」などの項目を属性(アトリビュート)と呼びます。MySQLなどでは、リレーションはテーブルと呼び、属性はフィールドあるいはカラムと呼びます。また、1行のデータをレコードと呼びます。たとえば、学生テーブルの学籍番号09R001の学生の姓は帝京、名は太郎ということを表す1行が1つのレコードです。

上で示したデータは、以下のような1つのテーブルで管理することもできます。このテーブルの方がシンプルに見えるかもしれませんが、この管理方法では、科目の担当者が変更になった場合に、複数のレコードを修正しなければなりません。上の3つのテーブルを使った管理方法では1つのレコードを修正するだけで済みます。

学籍番号	姓	名	科目名	担当教員
09R001	帝京	太郎	情報科学演習1	わたなべ
09R002	理工	次郎	情報科学演習1	わたなべ
09R001	帝京	太郎	情報科学演習2	ささき
08R001	通信	三郎	情報科学演習3	ささき
08R001	通信	三郎	情報科学演習4	わたなべ

どのようにテーブルを設計するかに関してはデータを管理しやすくするための方法論として、データベースの正規化がありますが、ここでは深入りしないことにします。

カラムには型(データタイプ)を指定します。MySQLでは以下のような型があります。

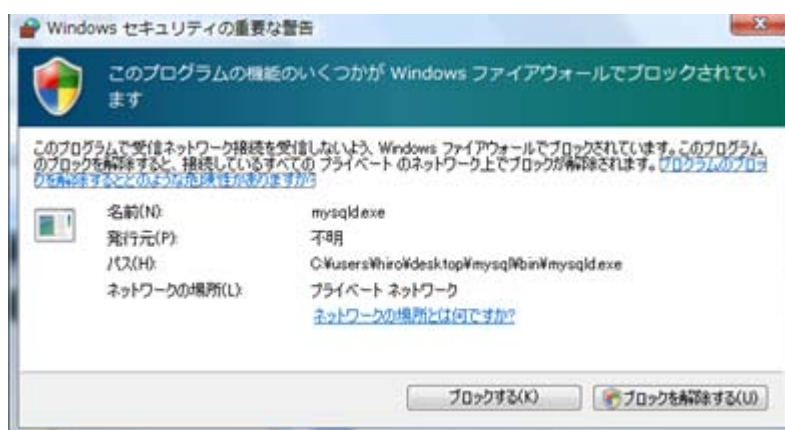
• CHAR(文字数)	固定桁数のデータ
• VARCHAR(文字数)	可変桁数のデータ
• TEXT	最大65535文字のテキスト
• LONGTEXT	最大約4.2G文字のテキスト
• TINYINT	とても小さい符号付整数 -128から127まで
• INT	通常の符号付整数 約 $\pm 2.1G$ までの数
• FLOAT	単精度浮動小数点数
• DOUBLE	倍精度浮動小数点数
• DATETIME	日時
• DATE	日付
• TIME	時間

MySQLの起動と停止

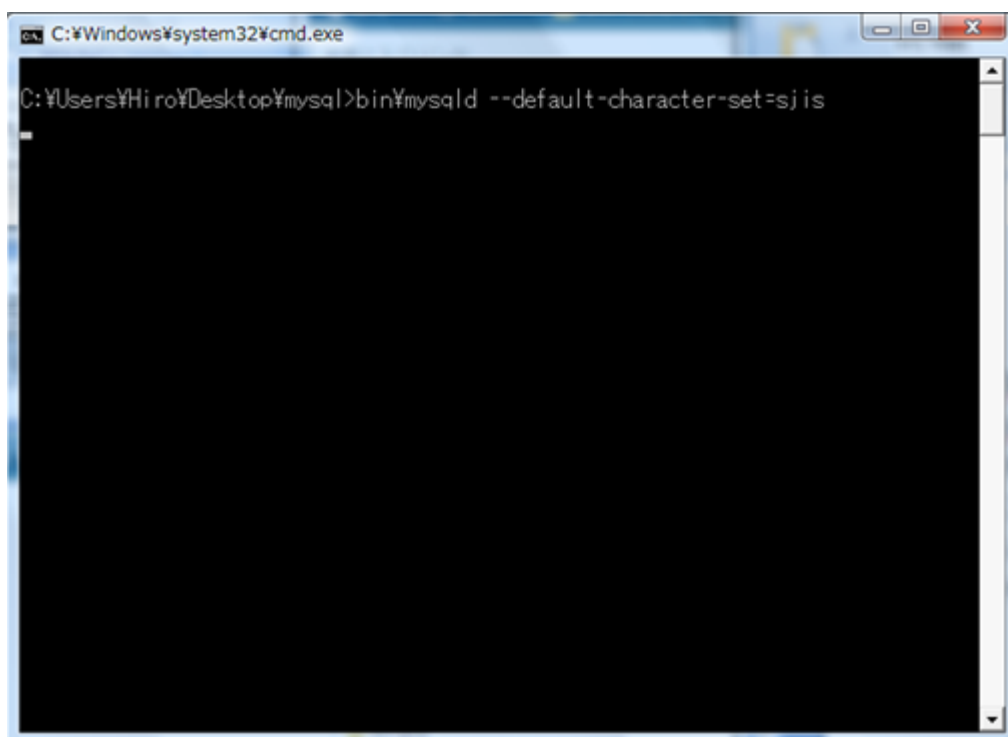
MySQLを使用して実習を行う際には、MySQLのサーバ(デーモン)を起動する必要があります。MySQLのインストール時に参照したウェブサイト等を参考にして、MySQLの起動方法と停止方法を確認しておきましょう。

授業のサイトからダウンロードしたmysqlの起動

この授業のサイトからダウンロードした場合は、mysql というフォルダの中に、mysql_start.bat というファイルがあるので、これをダブルクリックして起動します。その際に、以下のようなメッセージが表示される場合がありますが、ブロックを解除して実行します。



MySQLが起動すると、以下のようなウィンドウが出た状態になります。



このウィンドウは消さずに、表示されている状態か、最小化した状態で、データベースの実習を行います。

授業のサイトからダウンロードしたmysqlの停止

mysql というフォルダの中に、mysql_stop.batというファイルがあるので、これをダブルクリックすれば、mysqlが停止します。

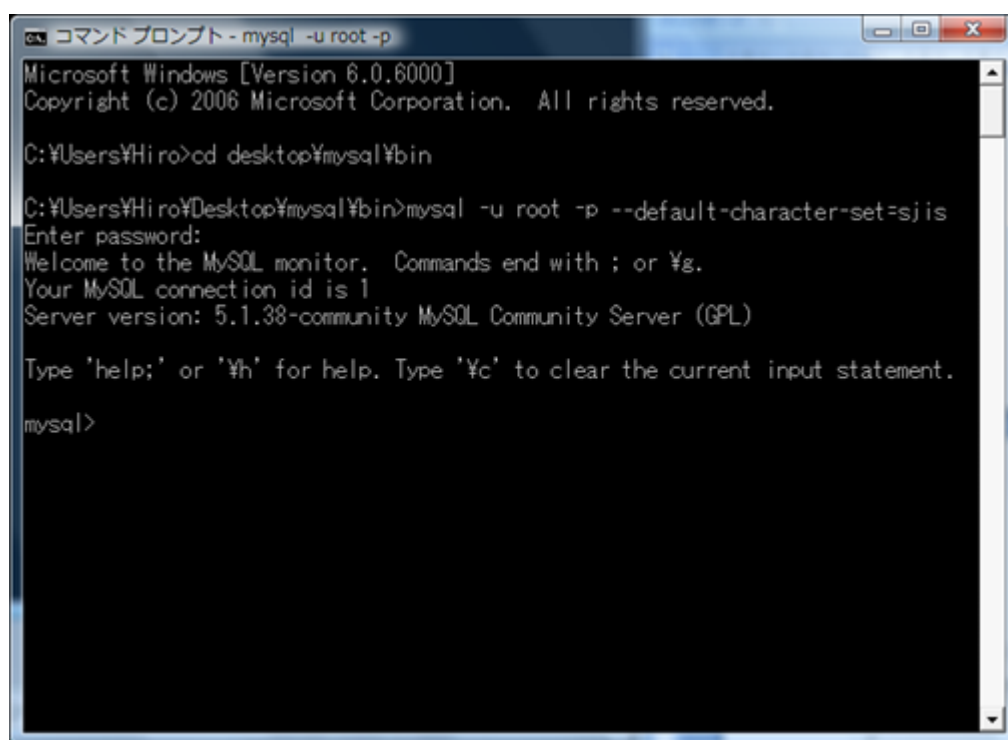
SQLモニタの利用

MySQLに直接コマンドを入力する場合には、SQLモニタを使います。以下の手順で起動します。

- (1) コマンドプロンプト を立ち上げます。コマンドプロンプトはWindowsのスタートボタンから、「すべてのプログラム」の「アクセサリ」の中にあります。
- (2) mysqlのフォルダの下にあるbinというフォルダにcdコマンドで移動します。
- (3) `mysql -u root -p --default-character-set=sjis` というコマンドを入力します。パスワードを聞かれるので、インストールした際に設定したパスワードを入力します。授業のサイトからダウンロードした場合は、パスワードは設定していませんので、そのままEnterキーを押してください。

プロンプトがmysql> となっている状態が、SQLモニタが動いている状態で、SQLのコマンドを入力することができます。

以下の例では、授業のサイトからダウンロードしたmysql をデスクトップにしています。そのため、まず、`cd desktop\mysql\bin` を入力することで、mysql の下のbin というフォルダに移動しました。次に`mysql -u root -p --default-character-set=sjis`を入力しています。



```
コマンドプロンプト - mysql -u root -p
Microsoft Windows [Version 6.0.6000]
Copyright (c) 2006 Microsoft Corporation. All rights reserved.

C:\Users\Hiro>cd desktop\mysql\bin

C:\Users\Hiro\Desktop\mysql\bin>mysql -u root -p --default-character-set=sjis
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1
Server version: 5.1.38-community MySQL Community Server (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

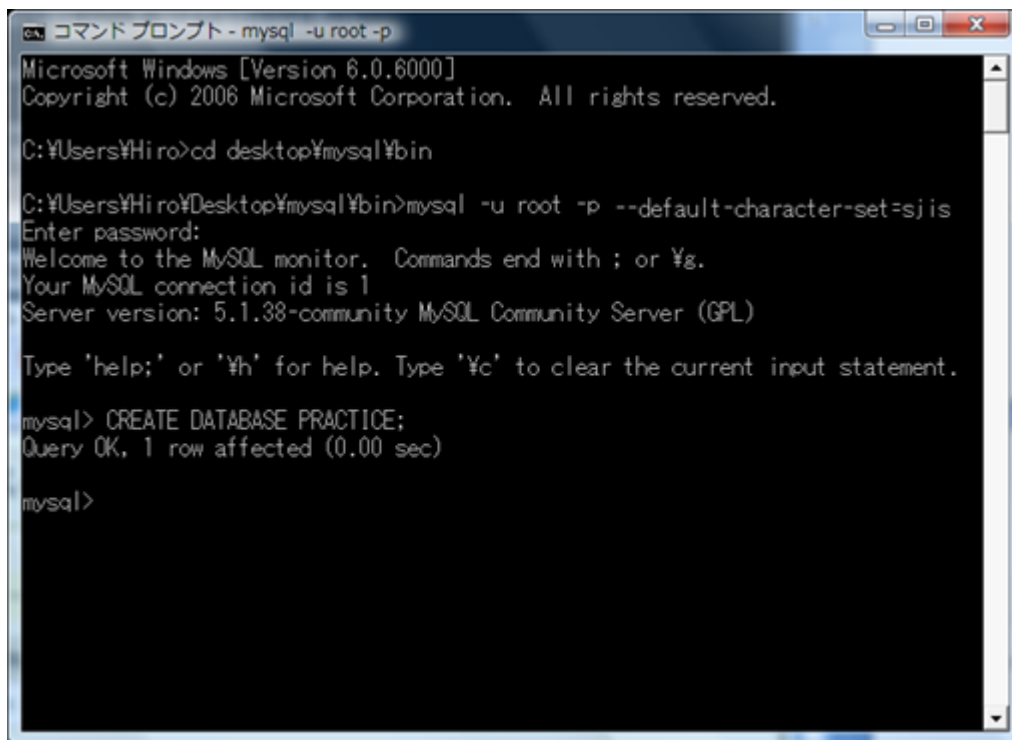
SQLモニタから抜けるときには、`exit` と入力し、Enterキーを押します。

データベースの作成

まず、データベースを作成してみましょう。データベースは、SQLモニタを起動して、以下のようなコマンドを実行します。MySQLでは、コマンドは大文字でも小文字でも実行できます。また、MySQLのコマンドの最後にはセミコロンをつけることも覚えておきましょう。

```
CREATE DATABASE データベース名;
```

以下の画面は、SQLモニタを起動した後、PRACTICE という名前のデータベースを作成した例です。



```
コマンドプロンプト - mysql -u root -p
Microsoft Windows [Version 6.0.6000]
Copyright (c) 2006 Microsoft Corporation. All rights reserved.

C:\Users\Hiro>cd desktop\mysql\bin

C:\Users\Hiro\Desktop\mysql\bin>mysql -u root -p --default-character-set=sjis
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1
Server version: 5.1.38-community MySQL Community Server (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE DATABASE PRACTICE;
Query OK, 1 row affected (0.00 sec)

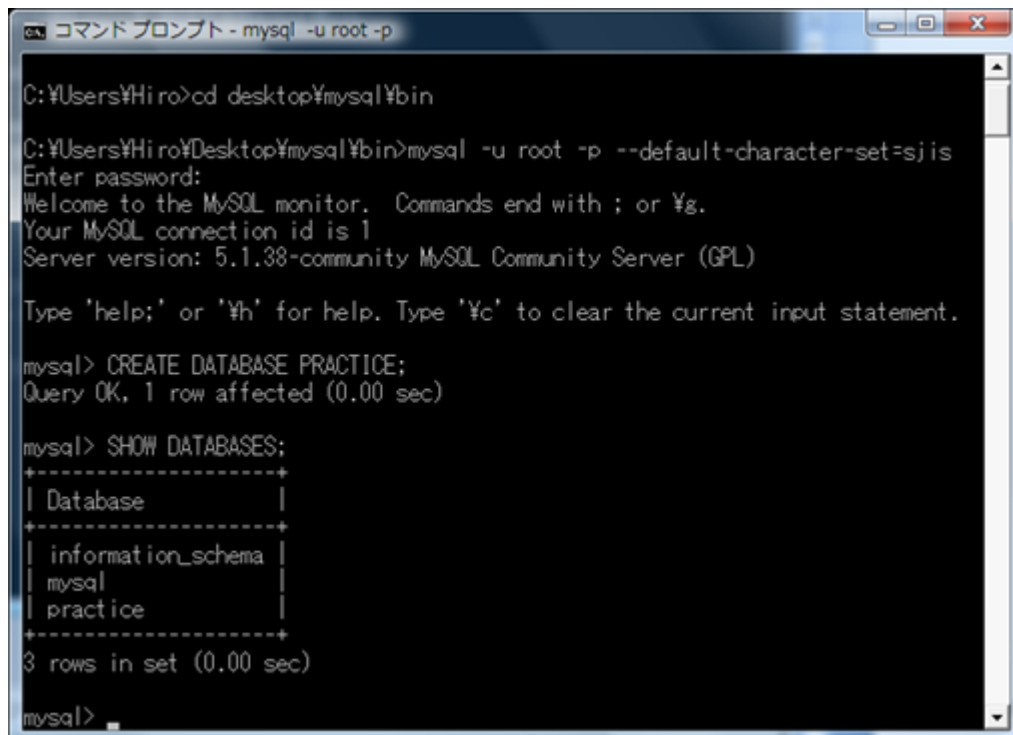
mysql>
```


データベースの一覧

どのようなデータベースがあるか、その一覧を見る場合は、以下のコマンドを実行します。

```
SHOW DATABASES;
```

以下の画面は PRACTICEというデータベース作成に引き続いて、データベースの一覧を見るコマンドを実行した例です。information_schema というデータベースとmysqlというデータベースは最初から存在します。その下にpracticeというデータベースがあることがわかります。データベースの名称も大文字・小文字の区別がないので、大文字で入力してデータベースを作成しても、小文字で表示されます。



```
コマンドプロンプト - mysql -u root -p

C:\Users\Hiro>cd desktop\mysql\bin

C:\Users\Hiro\Desktop\mysql\bin>mysql -u root -p --default-character-set=sjis
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1
Server version: 5.1.38-community MySQL Community Server (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE DATABASE PRACTICE;
Query OK, 1 row affected (0.00 sec)

mysql> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| mysql      |
| practice   |
+-----+
3 rows in set (0.00 sec)

mysql>
```

使用するデータベースの指定

使用するデータベースを指定するには以下のコマンドを実行します。

```
USE データベース名;
```

テーブル名を指定する際には、データベース名.テーブル名となりますが、このコマンドを実行しておくことで、テーブル名だけで指定することができます。

テーブルの作成

テーブルを作成するには以下のコマンドを実行します。

```
CREATE TABLE テーブル名 ( カラムの定義を必要なだけカンマで区切って記述 ) ;
```

カラムの定義は、カラム名 型を記述します。必要な場合は、オプションも指定します。また、プライマリキーをどれにするかといった指定も、カラムの定義の部分に記述します。以下に具体例を示します。

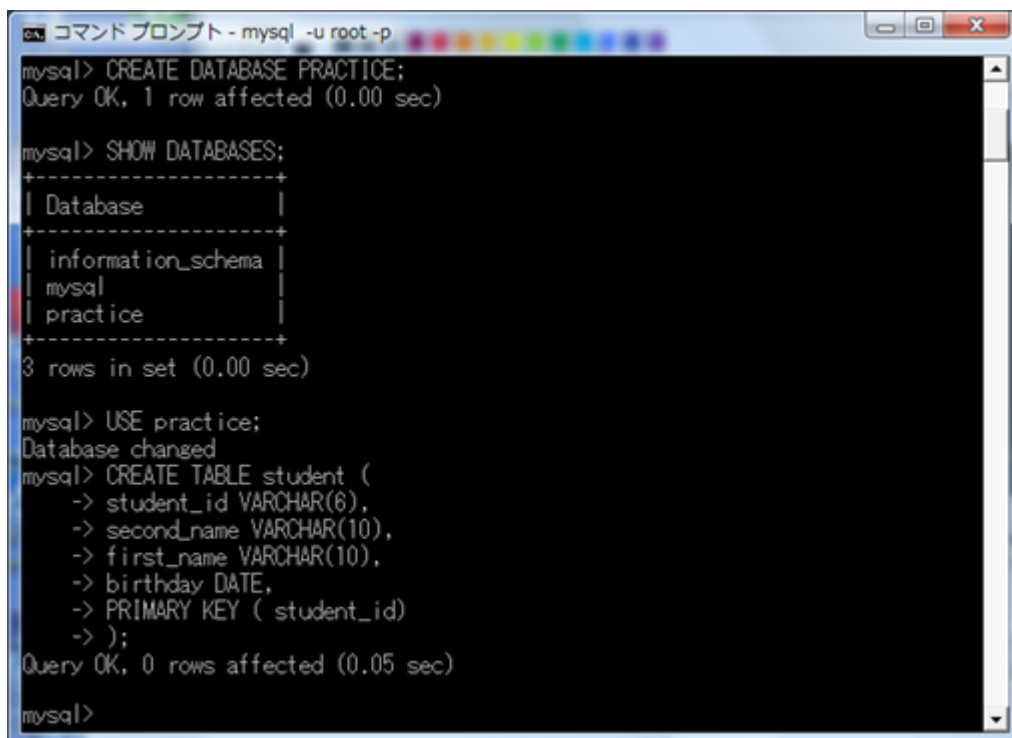
例 1

```
CREATE TABLE student ( student_id VARCHAR(6), second_name  
VARCHAR(10),first_name VARCHAR(10), birthday DATE, PRIMARY KEY ( student_id) )  
;
```

例 1 は、テーブル名が student で、以下のカラムを持ち、プライマリキーが student_id であるようなテーブルを作成するコマンドです。

カラム名	意味	データ型
student_id	学生ID	VARCHAR(6) 6文字以内の文字列
second_name	姓	VARCHAR(10) 10文字以内の文字列
first_name	名	VARCHAR(10) 10文字以内の文字列
birthday	誕生日	DATE 日付型

以下の画面は、データベースpracticeを使用することを指定した後、例1で示したstudentというテーブルを作成している例です。



```
mysql> CREATE DATABASE PRACTICE;  
Query OK, 1 row affected (0.00 sec)  
  
mysql> SHOW DATABASES;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| mysql |  
| practice |  
+-----+  
3 rows in set (0.00 sec)  
  
mysql> USE practice;  
Database changed  
mysql> CREATE TABLE student (  
-> student_id VARCHAR(6),  
-> second_name VARCHAR(10),  
-> first_name VARCHAR(10),  
-> birthday DATE,  
-> PRIMARY KEY ( student_id)  
-> );  
Query OK, 0 rows affected (0.05 sec)  
  
mysql>
```

SQLモニタでコマンドを入力する際に、改行をしてもかまいません。その際には、プロンプトが -> となります。もし入力ミスがあって、エラーが出た場合は最初から入力し直します。

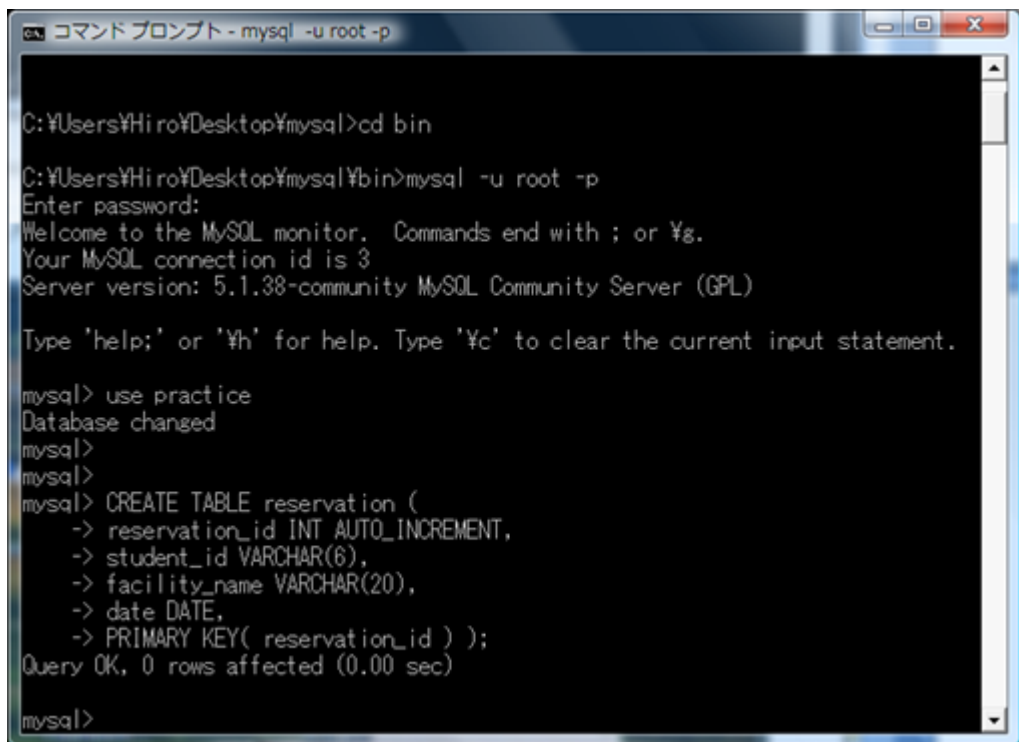
例 2

```
CREATE TABLE reservation ( reservation_id INT AUTO_INCREMENT , student_id
VARCHAR(6), facility_name VARCHAR(20), date DATE, PRIMARY KEY (
reservation_id) );
```

例2は、テーブル名が reservation で、以下のカラムを持ち、プライマリキーが reservation_id であるようなテーブルを作成するコマンドです。特に AUTO_INCREMENT は reservation_id には自動的に一意な値が設定されることを示しています。

カラム名	意味	データ型
reservation_id	予約ID	INT 整数型 自動割り当て
student_id	学生ID	VARCHAR(6) 6文字以内の文字列
facility_name	施設名	VARCHAR(20) 20文字以内の文字列
date	予約日	DATE 日付型

以下の画面は、SQL モニタを起動して、practice というデータベースを指定した後、例2で示したコマンドを実行した例です。



```
コマンドプロンプト - mysql -u root -p

C:\Users\Hiro\Desktop\mysql>cd bin

C:\Users\Hiro\Desktop\mysql\bin>mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3
Server version: 5.1.38-community MySQL Community Server (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use practice
Database changed
mysql>
mysql>
mysql> CREATE TABLE reservation (
-> reservation_id INT AUTO_INCREMENT,
-> student_id VARCHAR(6),
-> facility_name VARCHAR(20),
-> date DATE,
-> PRIMARY KEY( reservation_id ) );
Query OK, 0 rows affected (0.00 sec)

mysql>
```

テーブルの確認

テーブルの一覧

そのデータベースに存在するテーブルを一覧表示させるためには、以下のコマンドを実行します。

```
SHOW TABLES;
```

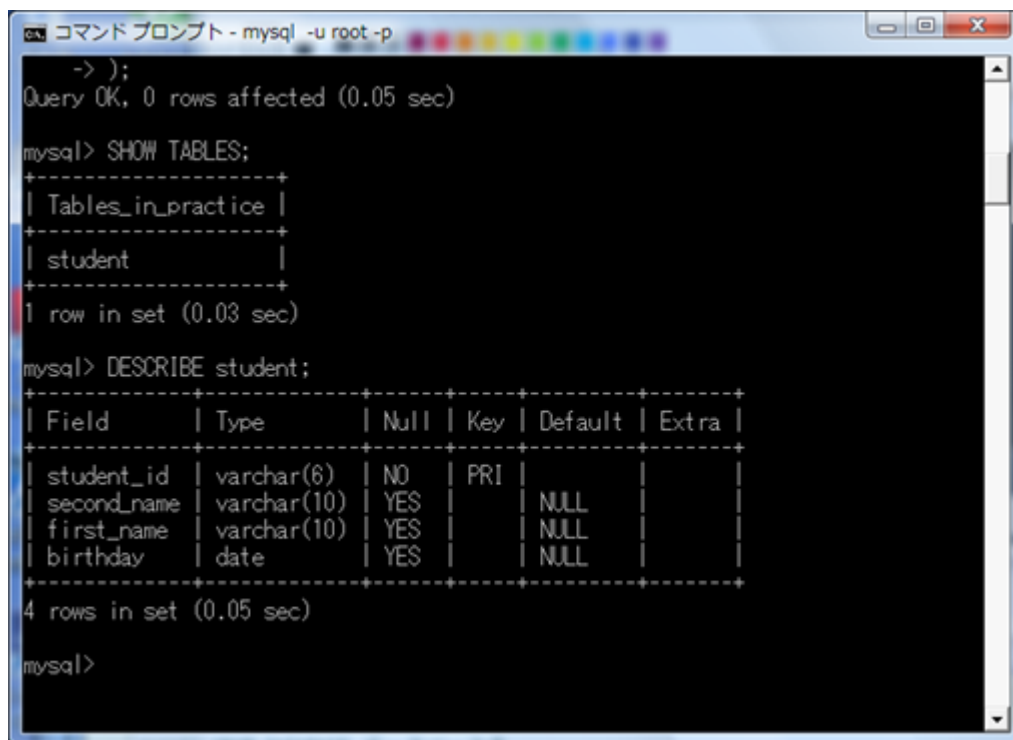
テーブルのカラムの確認

あるテーブルを構成するカラム名とそのデータ型を知りたいとき、以下のコマンドで表示させることができます。

```
DESCRIBE テーブル名;
```

実行例

以下の画面は、SHOW TABLESを実行し、次にDESCRIBE student を実行した例です。



```
コマンドプロンプト - mysql -u root -p
-> );
Query OK, 0 rows affected (0.05 sec)

mysql> SHOW TABLES;
+-----+
| Tables_in_practice |
+-----+
| student            |
+-----+
1 row in set (0.03 sec)

mysql> DESCRIBE student;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| student_id | varchar(6) | NO   | PRI |          |       |
| second_name | varchar(10) | YES  |     | NULL    |       |
| first_name  | varchar(10) | YES  |     | NULL    |       |
| birthday   | date      | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.05 sec)

mysql>
```

データベースとテーブルの削除

データベースの削除

間違った名称でデータベースを作ってしまった時など、データベースを削除したい場合には以下のコマンドを実行します。

```
DROP DATABASE データベース名;
```

このコマンドを実行すると、そのデータベースに含まれるテーブルとそこに格納されているデータが全て削除されて、元に戻すことはできません。データベースを削除するときには、十分に注意をして実行するようにしましょう。

テーブルの削除

同様に、テーブルを削除したいときは、以下のコマンドを実行します。

```
DROP TABLE テーブル名;
```

このコマンドを実行すると、そのテーブルに格納されているデータが全て削除されます。テーブルの削除の際には、十分に注意して実行しましょう。

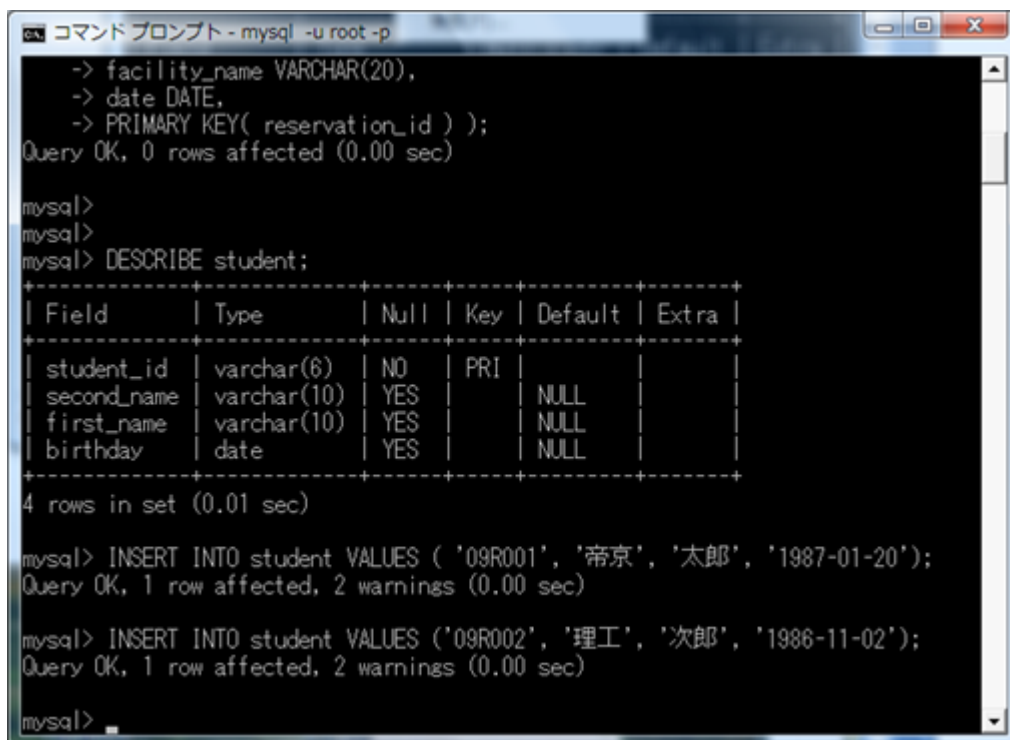
データの挿入

テーブルにデータを挿入するためには、以下のコマンドを実行します。

```
INSERT INTO テーブル名 VALUES ( 値をカンマで区切って記述 );
```

値をカンマで区切って記述する部分は、CREATE TABLEでテーブルを作った時に記述した順序にする必要があります。どんな順序で記述したか忘れた場合は、DESCRIBEコマンドを使って、テーブルの列を確認すれば順番がわかります。

以下の画面は、テーブル student に学生ID 09R001の帝京太郎と、09R002の理工次郎を挿入した例です。文字列は' (シングルクォーテーション) で囲むことに注意しましょう。また、DATE型の誕生日の記述方法も確認しておいてください。



```
コマンドプロンプト - mysql -u root -p
-> facility_name VARCHAR(20),
-> date DATE,
-> PRIMARY KEY( reservation_id );
Query OK, 0 rows affected (0.00 sec)

mysql>
mysql>
mysql> DESCRIBE student;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| student_id | varchar(6) | NO   | PRI |          |       |
| second_name | varchar(10) | YES  |     | NULL    |       |
| first_name  | varchar(10) | YES  |     | NULL    |       |
| birthday    | date       | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.01 sec)

mysql> INSERT INTO student VALUES ( '09R001', '帝京', '太郎', '1987-01-20' );
Query OK, 1 row affected, 2 warnings (0.00 sec)

mysql> INSERT INTO student VALUES ( '09R002', '理工', '次郎', '1986-11-02' );
Query OK, 1 row affected, 2 warnings (0.00 sec)

mysql>
```

Windowsのコマンドプロンプトで日本語が入力するには、[Alt]キーを押しながら、[半角/全角 漢字]キーを押すことで、日本語入力モードに切り替えます。

データの挿入方法として、全ての列には値を入れる必要がない場合は、以下のように最初に列名の列を記述してから、それに対応する値の列を記述する方法があります。

```
INSERT INTO テーブル名 ( カラム名をカンマで区切って記述 ) VALUES ( 値をカンマで区切って記述 );
```

カラム名の列と値の列の順番がきちんと対応するように記述することに注意しましょう。

以下の画面は、学生ID 09R001の学生がMeetingRoom01を2009年11月15日に予約したというレコードと、学生ID 09R002の学生がMeetingRoom02を2009年11月12日に予約したというレコードを挿入している様子を示しています。カラムreservation_idは、自動的に付与されるので、INSERT文では記入していないことに注意しましょう。また、1つめのレコードの挿入時と2つめのレコードの挿入時で、カラムの順番が異なりますが、それぞれのINSERT文でカラムの順番と値の順番はきちんと対応していることも確認しておきましょう。

```
コマンドプロンプト - mysql -u root -p

mysql>
mysql>
mysql> DESCRIBE reservation;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| reservation_id | int(11)       | NO   | PRI | NULL    | auto_increment |
| student_id     | varchar(6)    | YES  |     | NULL    |                |
| facility_name  | varchar(20)   | YES  |     | NULL    |                |
| date           | date          | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> INSERT INTO reservation ( student_id, facility_name, date )
-> VALUES ( '09R001', 'MeetingRoom01', '2009-11-15');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO reservation ( student_id, date, facility_name )
-> VALUES ( '09R002', '2009-11-12', 'MeetingRoom02');
Query OK, 1 row affected (0.00 sec)

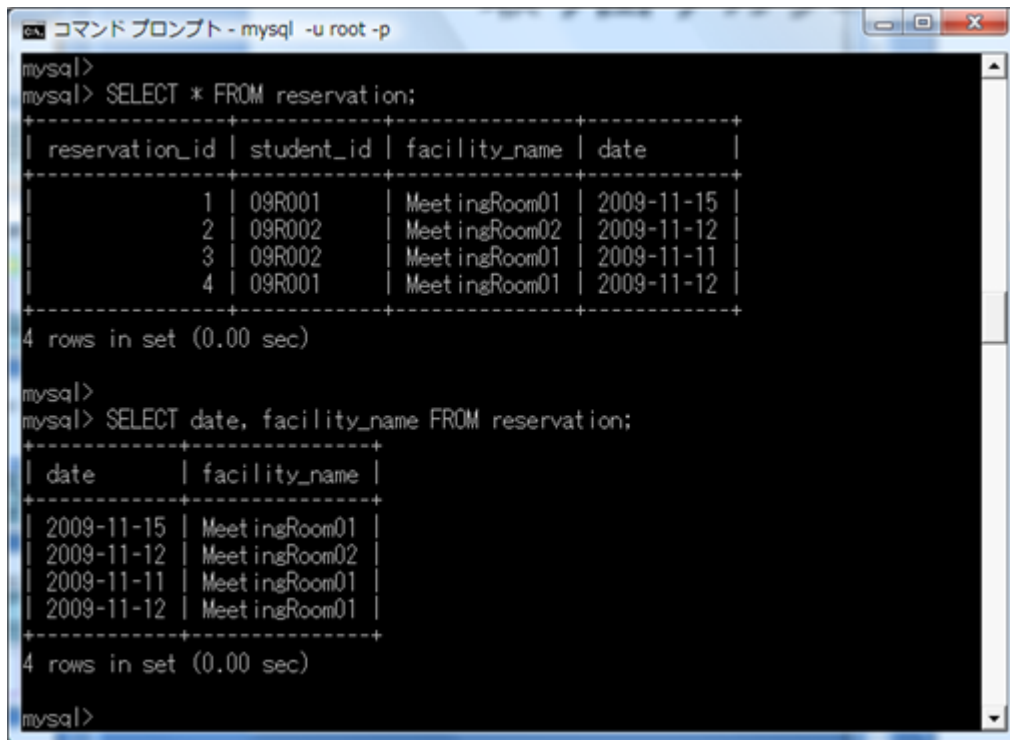
mysql>
```


データの検索

テーブルに格納されているデータを検索して表示するにはSELECT文を使います。
テーブルに格納されているレコードをすべて表示するには、以下のコマンドを実行します。

```
SELECT * FROM テーブル名
```

表示するカラムを指定するためには、*ではなく、カラム名をカンマで区切って記述します。以下の画面は、reservationテーブルから全てのレコードを表示した例とカラムを指定して表示した例です。



```
mysql>
mysql> SELECT * FROM reservation;
+-----+-----+-----+-----+
| reservation_id | student_id | facility_name | date       |
+-----+-----+-----+-----+
| 1              | 09R001     | MeetingRoom01 | 2009-11-15 |
| 2              | 09R002     | MeetingRoom02 | 2009-11-12 |
| 3              | 09R002     | MeetingRoom01 | 2009-11-11 |
| 4              | 09R001     | MeetingRoom01 | 2009-11-12 |
+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql>
mysql> SELECT date, facility_name FROM reservation;
+-----+-----+
| date       | facility_name |
+-----+-----+
| 2009-11-15 | MeetingRoom01 |
| 2009-11-12 | MeetingRoom02 |
| 2009-11-11 | MeetingRoom01 |
| 2009-11-12 | MeetingRoom01 |
+-----+-----+
4 rows in set (0.00 sec)

mysql>
```

表示するレコードの順序をあるカラムの値で並べ替えたいときには、ORDER BY カラム名を最後につけます。以下の画面は、予約日(DATE)で並べ替えて表示した例と学生IDで並べ替えて表示した例です。

```
コマンドプロンプト - mysql -u root -p
4 rows in set (0.00 sec)

mysql> SELECT * from reservation ORDER BY date;
+-----+-----+-----+-----+
| reservation_id | student_id | facility_name | date       |
+-----+-----+-----+-----+
| 3 | 09R002 | MeetingRoom01 | 2009-11-11 |
| 2 | 09R002 | MeetingRoom02 | 2009-11-12 |
| 4 | 09R001 | MeetingRoom01 | 2009-11-12 |
| 1 | 09R001 | MeetingRoom01 | 2009-11-15 |
+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> SELECT * from reservation ORDER BY student_id;
+-----+-----+-----+-----+
| reservation_id | student_id | facility_name | date       |
+-----+-----+-----+-----+
| 1 | 09R001 | MeetingRoom01 | 2009-11-15 |
| 4 | 09R001 | MeetingRoom01 | 2009-11-12 |
| 2 | 09R002 | MeetingRoom02 | 2009-11-12 |
| 3 | 09R002 | MeetingRoom01 | 2009-11-11 |
+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql>
```

あるカラムの値を条件として、条件を満たすレコードだけを表示するためには、WHERE を使って検索条件を指定します。検索条件はANDやORで複数の条件を組み合わせることができます。以下の画面は、学生IDが09R001のレコードを検索して表示した例と学生IDが09R002で予約日が2009年11月12日のレコードを検索して表示した例です。

```
コマンドプロンプト - mysql -u root -p

mysql>
mysql>
mysql>
mysql>
mysql>
mysql> SELECT * from reservation WHERE student_id='09R001';
+-----+-----+-----+-----+
| reservation_id | student_id | facility_name | date       |
+-----+-----+-----+-----+
| 1 | 09R001 | MeetingRoom01 | 2009-11-15 |
| 4 | 09R001 | MeetingRoom01 | 2009-11-12 |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> SELECT * from reservation WHERE student_id='09R002' AND date='2009-11-12';
+-----+-----+-----+-----+
| reservation_id | student_id | facility_name | date       |
+-----+-----+-----+-----+
| 2 | 09R002 | MeetingRoom02 | 2009-11-12 |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

まとめ

このモジュールでは以下のことを学びました。

- 関係データベースとその操作言語SQL
 - 関係データベースにおけるテーブル、カラムとそのデータ型
 - MySQLにおけるデータベース作成
 - MySQLにおけるテーブルの作成と確認、削除
 - MySQLにおけるデータの挿入
 - MySQLにデータ検索
-

課題7

(1) MySQLで、practice2というデータベースを作成し、以下のような3つのテーブルを作成しなさい。カラム名は英語、またはローマ字で表現し、半角英字で定義すること。さらに、それぞれの図で表されているレコードを挿入しなさい。

学生(Student)

学籍番号	姓	名
09R001	帝京	太郎
09R002	理工	次郎
08R001	通信	三郎

科目(Subject)

科目ID	科目名	担当教員
k001	情報科学演習1	わたなべ
k002	情報科学演習2	ささき
k003	情報科学演習3	ささき
k004	情報科学演習4	わたなべ

履修登録(Registration)

科目ID	学籍番号
k001	09R001
k001	09R002
k002	09R001
k003	08R001
k004	08R001

以上の操作が完了したら、それぞれのテーブルに対して、SELECT文で全てのレコードを表示して、表示された画面をキャプチャしておくこと。

(2) 第6回で作成した分析レベルのクラス図を基に、テーブルを設計しなさい。

MySQLにdb_reservation というデータベースを作成して、設計したテーブルを作成しなさい。設計したそれぞれのテーブルに対してDESCRIBE文でテーブルのカラムを表示して、表示された画面をキャプチャしておくこと。

(1)と(2)それぞれに対して、キャプチャした画面をワードなどに貼り付けて1つのファイルに収めて提出しなさい。
