#### Creating and Deleting a Database - CREATE DATABASE and DROP DATABASE

You can create a new database using SQL command "CREATE DATABASE databaseName"; and delete a database using "DROP DATABASE databaseName". You could optionally apply condition "IF EXISTS" or "IF NOT EXISTS" to these commands. For example,

mysql> **CREATE DATABASE southwind;**

**Use any name for database name other than southwind. You can use the same name for tables as given in help sheets.**

mysql> **DROP DATABASE southwind;**

mysql> **CREATE DATABASE IF NOT EXISTS southwind;**

mysql> **DROP DATABASE IF EXISTS southwind;**

**IMPORTANT**: Use SQL DROP (and DELETE) commands with extreme care, as the deleted entities are irrecoverable. **THERE IS NO UNDO!!!**

#### Creating and Deleting a Table - CREATE TABLE and DROP TABLE

You can create a new table in the default database using command "CREATE TABLE tableName" and "DROP TABLE tableName". You can also apply condition "IF EXISTS" or "IF NOT EXISTS". To create a table, you need to define all its columns, by providing the columns' name, type, and attributes.

Let's create a table "products" in our database "southwind".

-- Remove the database "southwind", if it exists.

-- Beware that DROP (and DELETE) actions are irreversible and not recoverable!

mysql> **DROP DATABASE IF EXISTS southwind;**

Query OK, 1 rows affected (0.31 sec)

-- Create the database "southwind"

mysql> **CREATE DATABASE southwind;**

Query OK, 1 row affected (0.01 sec)

-- Show all the databases in the server

-- to confirm that "southwind" database has been created.

mysql> **SHOW DATABASES;**

+--------------------+

| Database |

+--------------------+

| southwind |

| ...... |

-- Set "southwind" as the default database so as to reference its table directly.

mysql> **USE southwind;**

Database changed

-- Show the current (default) database

mysql> **SELECT DATABASE();**

+------------+

| DATABASE() |

+------------+

| southwind |

+------------+

-- Show all the tables in the current database.

-- "southwind" has no table (empty set).

mysql> **SHOW TABLES;**

Empty set (0.00 sec)

-- Create the table "products". Read "explanations" below for the column defintions

mysql> **CREATE TABLE products (**

**productID INT** **UNSIGNED** AUTO\_INCREMENT**,**

**productCode CHAR(3),**

**name VARCHAR(30),**

**quantity INT UNSIGNED,**

**price DECIMAL(7,2),**

**PRIMARY KEY (productID)**

**);**

Query OK, 0 rows affected (0.08 sec)

-- Show all the tables to confirm that the "products" table has been created

mysql> **SHOW TABLES;**

+---------------------+

| Tables\_in\_southwind |

+---------------------+

| products |

+---------------------+

-- Describe the fields (columns) of the "products" table

mysql> **DESCRIBE products;**

+-------------+------------------+------+-----+------------+----------------+

| Field | Type | Null | Key | Default | Extra |

+-------------+------------------+------+-----+------------+----------------+

| productID | int(10) unsigned | NO | PRI | NULL | auto\_increment |

| productCode | char(3) | NO | | | |

| name | varchar(30) | NO | | | |

| quantity | int(10) unsigned | NO | | 0 | |

| price | decimal(7,2) | NO | | 99999.99 | |

+-------------+------------------+------+-----+------------+----------------+

##### Explanations

We define 5 columns in the table products: productID, productCode, name, quantity and price. The types are:

* productID is INT UNSIGNED - non-negative integers.
* productCode is CHAR(3) - a fixed-length alphanumeric string of 3 characters.
* name is VARCHAR(30) - a variable-length string of up to 30 characters.  
  We use fixed-length string for productCode, as we assume that the productCode contains exactly 3 characters. On the other hand, we use variable-length string for name, as its length varies - VARCHAR is more efficient than CHAR.
* quantity is also INT UNSIGNED (non-negative integers).
* price is DECIMAL(7,2) - a decimal number with 2 decimal places.  
  DECIMAL is precise (represented as integer with a fix decimal point). On the other hand, FLOAT and DOUBLE (real numbers) are not precise and are approximated. DECIMAL type is recommended for currency..

We set the column productID as the so-called primary key. Values of the primary-key column must be unique. Every table shall contain a primary key. This ensures that every row can be distinguished from other rows. You can specify a single column or a set of columns (e.g., firstName and lastName) as the primary key. An index is build automatically on the primary-key column to facilitate fast search. Primary key is also used as reference by other tables.