

# Creating a Table By Hand

*Professor Don Colton, BYU Hawaii*

March 17, 2004

This handout is a shortened version of the full MySQL handout.

## 1 Connect to the MySQL Server

Use this command (e.g., from cs.byuh.edu) to connect to MySQL.

```
mysql -p -h HHH -u UUU
```

Pay attention to letter case. "ABC" is not the same as "Abc."

## 2 Focus on Your Database

Tell mysql to focus on your database by typing the following command:

```
mysql> use DDD;
```

where "DDD" is replaced by the name of the database assigned to you.

## 3 Databases Contain Tables

You will be working with tables within your database. You can see a list of the existing tables by using this command:

```
mysql> show tables;
```

Initially there should be nothing there. You should see a message saying something like "Empty set."

## 4 Task

Your task for this worksheet is to create an interesting table with at least three interesting columns and at least

three interesting rows. Things that are all the same are not interesting. Show some creativity. The following sections tell how to create the table, insert rows into it, and display the results.

## 5 Create a Table

Create a table by doing something like this:

```
mysql> create table scores (  
->   student varchar(50),  
->   score int(6)  
-> );
```

In this example, `scores` is the name of the table. It has two columns. One is called `student` and can hold a string of up to 50 characters. The second is called `score` and can hold a number up to nine digits, with a default printing width of six digits.

You can put that command all on one line if you like, or spread it out over multiple lines like shown above. mysql will continue prompting you for the remainder of your command until you put in the semi-colon ";" to tell it that you are done.

Now show tables again. You should see your new table.

## 6 Enter Data into Your Table

Insert something into your table by using commands like these.

```
insert into scores values ( "Fred", 100 );  
insert scores values ( "Bob", 70 );  
insert into scores ( score, student )  
  values ( 95, "Anne" );  
insert scores set student="Don", score=75;
```

## 7 Display the Data in Your Table

See what you have in your table by using a command like this. Star (\*) lists all columns in the default order, but you can assert more control if you want.

```
mysql> select * from scores;
mysql> select student from scores;
mysql> select score from scores;
mysql> select student, score from scores;
mysql> select score, student from scores;
```

You can also control the order in which your information is returned.

```
mysql> select * from scores order by score;
```

## 8 Column Data Types Allowed

In our example above, we created a table with two columns: student and score. Student was followed by the note “varchar(50)” and score was followed by the note “int(6)”. Varchar and int are called data types or column types. Here is a more complete sample of the column types allowed in MySQL. For a complete list, consult the MySQL book.

tinyint	-128 .. 127 (one byte)
smallint	-32768 .. 32767 (two bytes)
mediumint	-8388608 .. 8388607 (three bytes)
int	9 digits (four bytes)
bigint	20 digits (eight bytes)
float	like C, four bytes
double	like C, eight bytes
decimal(m,d)	string, m+2 bytes
char(m)	string, m bytes
varchar(m)	string, 1 to m+1 bytes
tinytext	up to 256 bytes
text	up to 65536 bytes
date	YYYY-MM-DD, three bytes
time	hh:mm:ss, three bytes
datetime	eight bytes
timestamp	four bytes (auto updating)
year	one byte

## 9 Updating a Row

You will need an update query. In this example, `mystuff` is a table name, `desc` is the column to be changed, `yadda` is a new value, and `ID` is the column to be matched.

```
update inven set desc="yadda" where ID=37;
update inven set desc="yadda"
  where ID=37 and price=33.91;
update inven set desc="yadda", price=99.99
  where ID=19;
```

## 10 Deleting a Row

You will need a delete query. In this example, `inventory` is a table name, `ID` is the column to be matched, and `99` is a value. If you leave off the “where” part, all rows in your table will be deleted.

```
delete from inventory where ID='99';
delete from inventory where ID like '201%';
delete from inventory where cost=price;
```

## 11 How to Add a Column

To modify an existing table, use the `alter table` query. Here are some samples of things you can do:

```
alter table foo add price int after cost;
alter table bar change price float;
alter table bleetch drop cost;
```

## 12 How to Delete a Table

To delete a table, use the `drop table` query.

```
drop table bleetch;
```