UHI Millennium Institute

Relational Database Management Systems

Standard Query Language

Drop Table

If you try and create a table in a database that already has that table in, an error will be returned. This is a good thing when you want to avoid duplicating your work or corrupting an already existing table. But what if you make a mistake and you want to start again. Or maybe you have created a command file and you need to get rid of the tables you created last time so as to run it again. This is where DROP TABLE comes in.

Note that there is no warning message that appears when you DROP TABLE and the data withing the table is lost for ever so make sure you think it through before using this SQL command.
The syntax for dropping a table is:

```
DROP TABLE [IF EXISTS] table_name;
```

So if we now look at how our command file might look when we add the DROP TABLE to the top so we can run this again and again it will look like this:

```
DROP TABLE IF EXISTS employee;
DROP TABLE IF EXISTS department;

CREATE TABLE department (
    departmentId INT           NOT NULL AUTO_INCREMENT,
    name         VARCHAR(30),
    PRIMARY KEY (departmentId)
) TYPE=INNODB;


CREATE TABLE employee (
    employeeId   INT           NOT NULL AUTO_INCREMENT,
    name         VARCHAR(80),
    job          VARCHAR(30),
    departmentId INT           NOT NULL,
    PRIMARY KEY (employeeId),
    FOREIGN KEY (departmentId) REFERENCES department(departmentId)
) TYPE=INNODB;
```

Note we have used the optional IF EXISTS. This ensures we do not get any errors if we are running this for the first time.

Note also the order of the DROP TABLE is the reverse of the table declaration. The order you drop tables is very import for the purposes of integrity. Because employee links to department if we get rid of department first employee will be linking to nothing and that is not allowed. Therefore we need to make sure we drop them in the right order.

Exercise:

Now we have seen how to create a table why not try it yourself. Type in the example given above and see if it runs and there are no syntax mistakes. Don't just cut and paste the text

but try typing it in yourself as this is good practice to get use to the syntax. Once you have it working, try running the query several times to check it really does work.

Now we want to see our tables are actually there and for this we use the SQL command show tables. The syntax of this is simply:

```
SHOW TABLES;
```

Finally try adding a new table called Address that separately stores the employees address. Populate the table with appropriate columns and their data types. The tables should be linked through employeeId.