

## Step 1: Logging In

- DB2 runs **only** on rees
- To log in to rees from another terminal window, you can type  
`rlogin rees`

## Step 2: Setting up Environment Variables

- For DB2 to function correctly, you need to set up certain environment variables
- Fortunately there are scripts that can do this for you
- The script must be run each time you log in to rees before using DB2. You can also automate this by calling the script from your startup script.
- The script you should use depends on what shell you are using. Type `echo $SHELL` to find out which shell you are using
  - If you are using sh or bash, type (note the leading period and space)  
`./u/cs448/public/db2profile`
  - If you are using csh or tcsh, type  
`source /u/cs448/public/db2cshrc`
- To automate this, place the appropriate command at **the end** of your `.profile` file (for bash & sh users) or `.cshrc` file (for csh and tcsh users)

### **Step 3: Starting DB2**

- If you have set up your environment correctly, simply typing `db2` should load DB2's interactive command facility

### **Step 4: Interacting with DB2**

There are three ways to send commands to DB2.

- Interactive Mode
  - After typing `db2` at the prompt, you should be in DB2's interactive mode. This mode presents you with a prompt and allows you to type commands one at a time and see the results of each command.
  - This mode is good for experimentation, but not as convenient for long series of queries.
- Batch Mode
  - To use this mode, you must first enter a sequence of DB2 commands into a text file (using a text editor)
  - Then, to have DB2 process the commands, you can type `db2 -f mycommands.clp`, where `mycommands.clp` is the text file containing your commands

## Step 5: Connecting to the CS 448 Database

- Before you can do anything useful in DB2, you must connect to a database, specifically, the CS 448 database
- To do this, you must use the command `connect to cs448` as your first command within DB2
- To access a table in the database you must precede all your table identifiers with the schema name
  - So, if I wanted to access the `department` table and the schema I am using is `db2_inst`, I would have to refer to the table as `db2_inst.department` in my query
  - To avoid having to precede all of your table identifiers with `db2_inst.`, you can instead enter the command `set schema db2_inst` and from that point on, DB2 will assume you are referring to the `db2_inst` schema whenever you use a table identifier.