Databases for Biologists

Session 3
Building And Modifying A Database With SQL

Robert Latek, Ph.D.
Sr. Bioinformatics Scientist
Whitehead Institute for Biomedical Research

Session 3 Outline

- SQL Query Review
- Creating Databases
- Creating Tables
- Altering Table Structure
- Inserting Data
- Deleting Data
- Updating/Modifying Data
- Automating Repetitive Tasks

Connecting To MySQL

- If No Local MySQL, In Terminal Window
  - % ssh hebrides.wi.mit.edu -l username
- Connect to MySQL Database Server
  - % mysql -u username -p -D db4bio
  - mysql>

- SQL Commands Are Case-Insensitive
- Tables And Attributes Are Case-Sensitive

SELECT

```
> SELECT *
  FROM Data
  LIMIT5;
```

affyId	exptId	level
AFFX-MurIL2_at	hs-cer-1	20
AFFX-MurIL10_at	hs-cer-1	8
AFFX-MurIL4_at	hs-cer-1	77
AFFX-MurFAS_at	hs-cer-1	30
AFFX-BioB-5_at	hs-cer-1	258

```
> SELECT DISTINCT species
  FROM LocusDescr;
```

species
Hs
Mm

WHERE And ORDER BY

```
> SELECT *
  FROM RefSeqs
  WHERE linkId BETWEEN 50 AND 100
  LIMIT 5;
```

linkId	ntRefSeq	aaRefSeq
50	NM_001098	NP_001089
51	NM_004035	NP_004026
52	NM_004300	NP_004291
53	NM_001610	NP_001601
54	NM_001611	NP_001602

```
> SELECT *
  FROM RefSeqs
  WHERE linkId BETWEEN 50 AND 100
  ORDER BY ntRefSeq DESC
  LIMIT5;
```

linkId	ntRefSeq	aaRefSeq
70	NM_005159	NP_005150
81	NM_004924	NP_004915
91	NM_004302	NP_004293
86	NM_004301	NP_004292
52	NM_004300	NP_004291

GROUP BY And HAVING

```
> SELECT *
  FROM Data
  GROUP BY affyId
  HAVING level < AVG(level)
  LIMIT 5;
```

affyId	exptId	level
100001_at	mm-hrt-1	5
100002_at	mm-hrt-1	20
100004_at	mm-hrt-1	154
100005_at	mm-hrt-1	660
100007_at	mm-hrt-1	585

```
> SELECT uId
  FROM UniSeqs
  GROUP BY uId
  HAVING count(gbId)>1
  LIMIT 5;
```

uId
Hs.100009
Hs.100014
Hs.100030
Hs.100058
Hs.100261

Table Joining

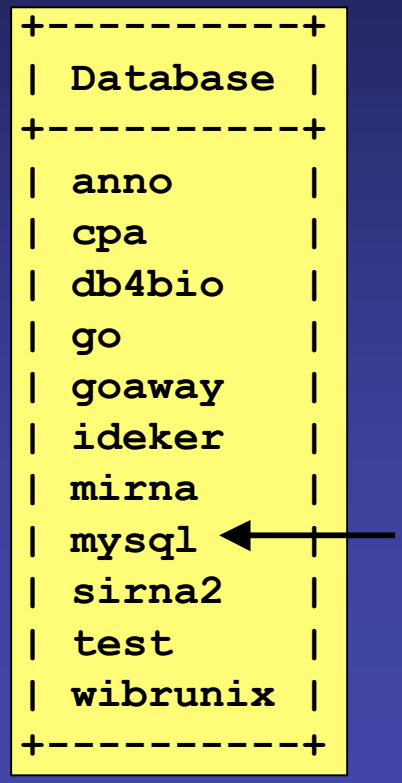
```
> SELECT DISTINCT Unigenes.uId, GO_Descr.description AS GO_description
  FROM Unigenes, LocusLinks, Ontologies, GO_Descr
  WHERE Unigenes.linkId=LocusLinks.linkId
  AND LocusLinks.linkId=Ontologies.linkId
  AND Ontologies.goAcc=GO_Descr.goAcc
  LIMIT 5;
```

uId	GO_description
Hs.373554	calcium ion binding
Hs.74561	protein carrier
Hs.155956	arylamine N-acetyltransferase
Hs.2	arylamine N-acetyltransferase
Hs.234726	serine protease inhibitor

CREATE DATABASE

- Allows You To Create A New Database On The Database Server

> SHOW DATABASES;
> CREATE DATABASE mfdb;
> SHOW DATABASES;
> USE mfdb;



Access Privileges

- Restrict Access And Prevent Accidental Alteration Of Important Information
- Can Limit What Individual Users Can See And Do On Particular Databases And Specific Tables
- Access Privileges Are Stored In The mysql Database

> GRANT ALL PRIVILEGES ON db4bio.* TO superuser@"%" IDENTIFIED BY "password";

> GRANT SELECT,INSERT ON db4bio.Data TO admin@"18.157.*.*" IDENTIFIED BY "pass2";

CREATE TABLE

- Translate An E-R Diagram (Schema) Into a Functioning Database


```
> CREATE TABLE Descriptions (
  gbId              VARCHAR(20)    NOT NULL,
  description       VARCHAR(100),
  PRIMARY KEY (gbId)
  );
```

```
+-------------+--------------+------+-----+---------+-------+
| Field       | Type         | Null | Key | Default | Extra |
+-------------+--------------+------+-----+---------+-------+
| gbId        | varchar(20)  |      | PRI |         |       |
| description | varchar(100) | YES  |     | NULL    |       |
+-------------+--------------+------+-----+---------+-------+
```

CREATE TABLE

Targets
affyId
gbId
species

```
> CREATE TABLE Targets (
  affyId          VARCHAR(20)   NOT NULL,
  gbId            VARCHAR(20)   NOT NULL,
  species         VARCHAR(20),
  PRIMARY KEY (affyId, gbId)
  );
```

```
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| affyId  | varchar(20) |      | PRI |         |       |
| gbId    | varchar(20) |      | PRI |         |       |
| species | varchar(20) | YES  |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
```

ALTER TABLE

- Modify A Table's Attributes
  - Attribute Names, Type, Null, Key, Default
  - Add Or Drop Attributes

```
> ALTER TABLE Data
  CHANGE level level DOUBLE;

> ALTER TABLE Data
  RENAME level expression;

> ALTER TABLE Data
  ADD PRIMARY KEY (exptId);
```

```
> ALTER TABLE Data
  DROP COLUMN affyId;

> ALTER TABLE Data
  ADD date TIMESTAMP;

> DROP TABLE Data;
```

INSERT INTO

- Finally, Add Data Into Tables

```
> INSERT INTO Data (level, exptId, affyId)
  VALUES (215, “hs-hrt-1”, “100008_at”);
```

EXPLICIT ORDER

```
> INSERT INTO Data
  VALUES (“100008_at”, “hs-hrt-1”, 215);
```

IMPLIED ORDER

```
> INSERT INTO Data2 (affyId2,level2)
  SELECT Data.affyId, Data.level
  FROM Data
  WHERE Data.level < 250;
```

DATA COPYING

DELETE FROM

- Delete Data From Tables
- Similar Syntax As SELECT

```
> DELETE FROM Data
  WHERE exptId="hs-hrt-1";

> DELETE FROM Sources
  WHERE exptId= "hs-hrt-1";
```

BE CONSISTENT

UPDATE

- Modify Data Already Stored In A Table
- Again, Similar Syntax As SELECT

```
> UPDATE Data                                   MODIFY
  SET exptId="hs-hrt-2"
  WHERE exptId="hs-hrt-1";

> UPDATE Source                                 FIX
  SET exptId= "ms-hrt-1", source="Mm"
  WHERE exptId="hs-hrt-1";

> UPDATE Data                                   INTERNAL
  SET level=level*2                             NORMALIZATION
  WHERE exptId="hs-hrt-1";
```

LOAD DATA And Export

- Read Rows From A Text File Into A Table And Vice Versa

```
> LOAD DATA LOCAL INFILE “data.txt”
  INTO TABLE db4bio.Data
  FIELDS TERMINATED BY ‘\t’ ENCLOSED BY ‘’ ESCAPED BY’\\’    (ASSUMED)
  LINES TERMINATED BY ‘\n’;                                  (ASSUMED)

> LOAD DATA LOCAL INFILE “data.txt”
  INTO TABLE db4bio.Data
  FIELDS TERMINATED BY ‘,’;

> SELECT * INTO OUTFILE “data.txt”
  FIELDS TERMINATED BY ‘,’
  FROM Data;
```

Automating Repetitive Tasks

- Use .SQL Files To Perform SQL Commands Automatically
- Automatically Create A Series Of Tables

```
% mysql -h hebrides.wi.mit.edu -u guest -p -D databasename < create.sql
```

- Feed A Complicated Query To The Database And Receive The Results In A Text File

```
% mysql -h hebrides.wi.mit.edu -u web -p -D db4bio < query1.sql > query1.out
```

Summary

- Design Databases With E-R Diagrams
- Data Mine Using Combinations Of SELECT/FROM With WHERE, GROUP BY, HAVING, ORDER BY, And Aggregates
- Create And Implement Databases
- Input and Output Data From Databases
- Modify Existing Data Within Databases

Where To Go From Here?

- Consult SQL And MySQL Resources
  - http://www.mysql.com
  - http://neo.bu.edu/be768/2003Class/
- Database Tools
  - VisualCase2 (draw E-R diagrams)
  - Data Architect (draw E-R diagrams)
  - Visio (PC - draw E-R diagrams & DB Administration)
  - SQL4XManagerJ (DB Administration)

Exercises

- Create Tables
- Input Data
- Modify/Delete Particular Data

- Accessing Your Database:
  - mysql -u *username* -p -D *username*