Databases for Biologists

Session 3
Building And Modifying
A Database With SQL

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

Connecting To MySQL

- If No Local MySQL, In Terminal Window
  - % ssh hebridess.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

WHERE And ORDER BY

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

<table>
<thead>
<tr>
<th>linkID</th>
<th>ntRefSeq</th>
<th>aaRefSeq</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>NM_001098</td>
<td>NP_001089</td>
</tr>
<tr>
<td>51</td>
<td>NM_004035</td>
<td>NP_004026</td>
</tr>
<tr>
<td>52</td>
<td>NM_004300</td>
<td>NP_004291</td>
</tr>
<tr>
<td>53</td>
<td>NM_004610</td>
<td>NP_004601</td>
</tr>
<tr>
<td>54</td>
<td>NM_001611</td>
<td>NP_001602</td>
</tr>
</tbody>
</table>

GROUP BY And HAVING

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

<table>
<thead>
<tr>
<th>affyId</th>
<th>expId</th>
<th>level</th>
</tr>
</thead>
<tbody>
<tr>
<td>100001_at</td>
<td>mm-bhr-1</td>
<td>154</td>
</tr>
<tr>
<td>100002_at</td>
<td>mm-bhr-1</td>
<td>160</td>
</tr>
<tr>
<td>100004_at</td>
<td>mm-bhr-1</td>
<td>585</td>
</tr>
</tbody>
</table>

> SELECT * FROM UniSeqs
WHERE gbId BETWEEN 50 AND 100
ORDER BY gbId DESC
LIMIT 5;

<table>
<thead>
<tr>
<th>gbId</th>
<th>gbId1</th>
<th>gbId2</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>51</td>
<td>52</td>
</tr>
<tr>
<td>52</td>
<td>53</td>
<td>54</td>
</tr>
</tbody>
</table>

Session 3 Outline

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

SELECT

> SELECT * FROM Data
LIMIT 5;

<table>
<thead>
<tr>
<th>affyId</th>
<th>expId</th>
<th>level</th>
</tr>
</thead>
<tbody>
<tr>
<td>100001_at</td>
<td>mm-bhr-1</td>
<td>5</td>
</tr>
<tr>
<td>100002_at</td>
<td>mm-bhr-1</td>
<td>20</td>
</tr>
<tr>
<td>100004_at</td>
<td>mm-bhr-1</td>
<td>154</td>
</tr>
<tr>
<td>100005_at</td>
<td>mm-bhr-1</td>
<td>660</td>
</tr>
<tr>
<td>100007_at</td>
<td>mm-bhr-1</td>
<td>585</td>
</tr>
</tbody>
</table>

> SELECT DISTINCT species FROM LocusDescr;

<table>
<thead>
<tr>
<th>species</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hs.</td>
</tr>
<tr>
<td>Mm.</td>
</tr>
</tbody>
</table>

> SELECT * FROM RefSeqs
WHERE linkID BETWEEN 50 AND 100
ORDER BY ntRefSeq DESC
LIMIT 5;

<table>
<thead>
<tr>
<th>linkID</th>
<th>ntRefSeq</th>
<th>aaRefSeq</th>
</tr>
</thead>
<tbody>
<tr>
<td>70</td>
<td>NM_003159</td>
<td>NP_003150</td>
</tr>
<tr>
<td>81</td>
<td>NM_004924</td>
<td>NP_004915</td>
</tr>
<tr>
<td>91</td>
<td>NM_004302</td>
<td>NP_004293</td>
</tr>
<tr>
<td>86</td>
<td>NM_004301</td>
<td>NP_004292</td>
</tr>
<tr>
<td>52</td>
<td>NM_004300</td>
<td>NP_004291</td>
</tr>
</tbody>
</table>
### Table Joining

```sql
> 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;
```

<table>
<thead>
<tr>
<th>uId</th>
<th>GO_description</th>
</tr>
</thead>
<tbody>
<tr>
<td>373554</td>
<td>calcium ion binding</td>
</tr>
<tr>
<td>74561</td>
<td>protein carrier</td>
</tr>
<tr>
<td>155956</td>
<td>arylamine N-acetyltransferase</td>
</tr>
<tr>
<td>2.2</td>
<td>asparagine N-acetyltransferase</td>
</tr>
<tr>
<td>234726</td>
<td>serine protease inhibitor</td>
</tr>
</tbody>
</table>

### CREATE DATABASE

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

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

### CREATE TABLE

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

```sql
> CREATE TABLE Descriptions(  
| gbid | description | NOT NULL,  
<table>
<thead>
<tr>
<th></th>
<th></th>
<th>PRIMARY KEY (gbid)</th>
</tr>
</thead>
</table>
```

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Key</th>
<th>Default</th>
<th>Extra</th>
</tr>
</thead>
<tbody>
<tr>
<td>gbid</td>
<td>varchar(20)</td>
<td></td>
<td>PRI</td>
<td></td>
<td></td>
</tr>
<tr>
<td>desc</td>
<td>varchar(100)</td>
<td>YES</td>
<td>NULL</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### ALTER TABLE

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

```sql
> ALTER TABLE Data
CHANGE level level DOUBLE;
> ALTER TABLE Data
DROP COLUMN affid;
> ALTER TABLE Data
RENAME level expression;
> ALTER TABLE Data
ADD PRIMARY KEY (exptid);
> ALTER TABLE Data
DROP TABLE Data;
```
Automating Repetitive Tasks

- Use .SQL Files To Perform SQL Commands Automatically
  
- Automatically Create A Series Of Tables
  
- Feed A Complicated Query To The Database And Receive The Results In A Text File

```
% mysql -h hebrides.wi.mit.edu -u guest -p -D databasename < create.sql
% 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://www.mysql.com)
  - [http://neo.bu.edu/be768/2003Class/](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`