SQL Practice
Schema

borrower

| Customer_name |
| loan_number |

loan

| loan_number |
| amount |
| branch_name |

branch

| branch_name |
| branch_city |
| assets |

depositor

account

| account_num |
| amount |
| branch_name |

| Customer_name |
| account_num |
Find the names of branches that have greater assets than some branch located in Brooklyn

```
select distinct T.branch_name 
from branch as T, branch as S 
where T.assets > S.assets 
and S.branch_city = 'Brooklyn'
```
String Operation (naïve text search)

- percent % or *: matches any substring
- underscore _ or ?: matches any character

Find the names of depositors whose name includes the substring “Mary”

- select customer_name
- from depositor
- where customer_name like ‘%Mary%’
String Operation (cont.)

- Any street name with exactly 5 characters
  - ... like ‘_ _ _ _ _’

- More string operation:
  - concatenation (using “||”)
  - finding string length, extracting substrings, etc.

- Full text search
  - Postgresql 8.3 also supports full text search
  - [http://www.postgresql.org/docs/8.3/static/textsearch.html](http://www.postgresql.org/docs/8.3/static/textsearch.html) for more details
Nested Query

- Find branches that have greater assets than some branch located in Brooklyn

```sql
select branch_name
from branch
where assets > some
  (select assets
   from branch
   where branch_city = 'Brooklyn')
```
Quantifier

- **Some**
  - $5 < \text{ some } \{0,5,6\}$? True
  - $5 < \text{ some } \{0,5\}$? False
  - $5 = \text{ some } \{0,5\}$? True
  - $5 \neq \text{ some } \{0,5\}$? True

- **All**
  - $5 < \text{ all } \{0,5,6\}$? False
  - $5 = \text{ all } \{4,5\}$? False
  - $5 \neq \text{ all } \{4,6\}$? True
Quantifier (cont.)

- Find customers who have an account at all branches located in Brooklyn

```sql
select d.customer_name 
From depositor d, account a 
Where d.account_num = a.account_num AND 
    a.branch_name = all 
    ( select branch_name 
        from branch 
        where branch_city = 'Brooklyn' 
    )
```

- Wrong!
correct solution

```sql
select distinct S.customer_name
from depositor as S
where not exists ( 
    (select branch_name
    from branch
    where branch_city = 'Brooklyn')
except
    (select R.branch_name
    from depositor as T, account as R
    where T.account_number = R.account_number and
    S.customer_name = T.customer_name )
)
```

- exists \( r: r \neq \emptyset \)
- not exists \( r: r = \emptyset \)
- \( X \) except \( Y: X-Y \)
MySQL

- All the DBs provide client terminal
- show databases
- Use DB_NAME
- show TABLE_NAME
MySQL

- CREATE TABLE person (pid SERIAL, firstname CHAR(15), lastname CHAR(15), PRIMARY KEY (pid,...))
- CREATE TABLE role (rid SERIAL, pid INTEGER REFERENCES person(pid), role char(10))
- INSERT INTO person VALUES (default, 'Jeff', 'Chen')
JDBC

- Java program to manipulate data in DB

```java
Connection conn = DriverManager.getConnection("jdbc:mysql://localhost/mydb?user=root");
Statement stmt = conn.createStatement();
ResultSet rs = stmt.executeQuery("select * from person");
While (rs.next()) {
    int id = rs.getInt("pid");
    String firstname = rs.getString("firstname");
}
```
JDBC

- Add xxxjdbc.jar to classpath

- Transaction
  
  ```java
  Statement stmt;
  Stmt.setAutoCommit(FALSE);
  Stmt.commit();
  ...
  ...
  Stmt.commit();
  ...
  ```

  One transaction
JDBC

Read more

- Java SQL API
- http://dev.mysql.com/usingmysql/java/
backup
More experience on SQL

☐ Ready ...
☐ Go~~~
1. which supplier should be selected to place an order for a given part in a given region?

**Select** ......

**From** part, supplier, partsupp, nation, region

**Where**

\[
\begin{align*}
p_{\text{partkey}} &= ps_{\text{partkey}} \\
\textbf{AND} &\ s_{\text{suppkey}} = ps_{\text{suppkey}} \\
\textbf{AND} &\ p_{\text{size}} = [\text{SIZE}] \\
\textbf{AND} &\ p_{\text{type}} \text{ like } '%[\text{TYPE}]' \\
\textbf{AND} &\ s_{\text{nationkey}} = n_{\text{nationkey}} \\
\textbf{AND} &\ n_{\text{regionkey}} = r_{\text{regionkey}} \\
\textbf{AND} &\ r_{\text{name}} = '[\text{REGION}]' \\
\textbf{AND} &\ ps_{\text{supplycost}} = ( \\
\end{align*}
\]
Select min(ps_supplycost)
From partsupp, supplier, nation, region
Where p_partkey = ps_partkey
AND s_suppkey = ps_suppkey
AND s_nationkey = n_nationkey
AND n_regionkey = r_regionkey
AND r_name = '[REGION]'
)
2. how well the order priority system is working?

- Counts the number of orders ordered in a given quarter of a given year in which at least one lineitem was received by the customer later than its committed date

```sql
Select o_orderpriority,
    count(*) as order_count
from orders
```
WHERE o_orderdate >= date '[DATE]' ' 
AND o_orderdate < date '[DATE]' + interval '3' month
AND exists ( 
    select * 
    From lineitem 
    where l_orderkey = o_orderkey 
    AND l_commitdate < l_receiptdate 
)
GROUP BY o_orderpriority
ORDER BY o_orderpriority;
3. who might be having problems with the parts that are shipped to them?

**Select**
c_custkey,
c_name,
sum(l_extendedprice * (1 - l_discount)) as revenue,
c_acctbal,
n_name,
c_address,
c_phone,
c_comment

**from**
customer,
orders,
lineitem,
nation
Where c_custkey = o_custkey and l_orderkey = o_orderkey and l_returnflag = 'R' and c_nationkey = n_nationkey
group by c_custkey,
    c_name,
    c_acctbal,
    c_phone,
    n_name,
    c_address,
    c_comment
order by revenue desc;