Lecture 2: Beginning SQL

Monday, January 25, 2015
Agenda for today

- Class Demo: Oracle set up for Mac using Virtual Box (Yuming)
- Brief overview of relational database systems
- SQL: Chapter 3 in Murach textbook
Market shares

From 2011 Gartner report:

- Oracle: 48% market with $11.7BN in sales
- IBM: 20% market with $4.8BN in sales
- Microsoft: 17% market with $4.0BN in sales
- Other Vendors: 5.8% market with $1.3BN in sales
**Basic concepts**

- Relational model
- Relation / Entity / Table
- Field / Attribute / Column
- Row / Tuple / Record
- Cell / Value
- Primary key
- Composite primary key
- Foreign key
- Constraint
The Vendors table in the Accounts Payable schema

<table>
<thead>
<tr>
<th>VENDOR_ID</th>
<th>VENDOR_NAME</th>
<th>VENDOR_ADDRESS1</th>
<th>VENDOR_ADDRESS2</th>
<th>VENDOR_CITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1US Postal Service</td>
<td>Attn: Supt. Window Services</td>
<td>PO Box 7005</td>
<td>Madison</td>
</tr>
<tr>
<td>2</td>
<td>2National Information Data Ctr</td>
<td>PO Box 96621</td>
<td>NULL</td>
<td>Washington</td>
</tr>
<tr>
<td>3</td>
<td>3Register of Copyrights</td>
<td>Library Of Congress</td>
<td>NULL</td>
<td>Washington</td>
</tr>
<tr>
<td>4</td>
<td>4Jobtrak</td>
<td>1990 Westwood Blvd Ste 260</td>
<td>NULL</td>
<td>Los Angeles</td>
</tr>
<tr>
<td>5</td>
<td>5Newbrige Book Clubs</td>
<td>3000 Cindel Drive</td>
<td>NULL</td>
<td>Washington</td>
</tr>
<tr>
<td>6</td>
<td>6California Chamber Of Commerce</td>
<td>3255 Ramos Cir</td>
<td>NULL</td>
<td>Sacramento</td>
</tr>
<tr>
<td>7</td>
<td>7Towne Advertiser's Mailing Svcs Kevin Minder</td>
<td>3411 W Macarthur Blvd</td>
<td>Santa Ana</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>8EFI Industries</td>
<td>PO Box 9365</td>
<td>NULL</td>
<td>Fresno</td>
</tr>
<tr>
<td>9</td>
<td>9Pacific Gas &amp; Electric</td>
<td>Box 52001</td>
<td>NULL</td>
<td>San Francisco</td>
</tr>
<tr>
<td>10</td>
<td>10Robbins Mobile Lock And Key</td>
<td>4669 N Fresno</td>
<td>NULL</td>
<td>Fresno</td>
</tr>
<tr>
<td>11</td>
<td>11Bill Marvin Electric Inc</td>
<td>4583 E Home</td>
<td>NULL</td>
<td>Fresno</td>
</tr>
<tr>
<td>12</td>
<td>12City Of Fresno</td>
<td>PO Box 2069</td>
<td>NULL</td>
<td>Fresno</td>
</tr>
<tr>
<td>13</td>
<td>13Golden Eagle Insurance Co</td>
<td>PO Box 85526</td>
<td>NULL</td>
<td>San Diego</td>
</tr>
<tr>
<td>14</td>
<td>14Expedata Inc</td>
<td>4420 N. First Street, Suite 108</td>
<td>NULL</td>
<td>Fresno</td>
</tr>
</tbody>
</table>
Tables Explained

- A tuple = a record
- A table = a set of records
- The schema of a table is the table name and attributes
- A key is an attribute whose value is unique (by convention, we underline the key)
The columns of the Invoices table
Common data types

- CHAR, VARCHAR2
- NUMBER
- FLOAT
- DATE
- BLOB, CLOB
Constraint types

- NOT NULL constraint
- Unique constraint
- Primary and foreign key constraint
- Check constraint
The relationship between two tables

Primary key

<table>
<thead>
<tr>
<th>VENDOR_ID</th>
<th>VENDOR_NAME</th>
<th>VENDOR_ADDRESS1</th>
</tr>
</thead>
<tbody>
<tr>
<td>113</td>
<td>114 Postmaster</td>
<td>Postal Due Technician</td>
</tr>
<tr>
<td>114</td>
<td>115 Roadway Package System, Inc</td>
<td>Dept La 21095</td>
</tr>
<tr>
<td>115</td>
<td>116 State of California</td>
<td>Employment Development Dept</td>
</tr>
<tr>
<td>116</td>
<td>117 Suburban Propane</td>
<td>2374 S Cherry Ave</td>
</tr>
<tr>
<td>117</td>
<td>118 Unocal</td>
<td>P.O. Box 660070</td>
</tr>
<tr>
<td>118</td>
<td>119 Yesmed, Inc</td>
<td>PO Box 2061</td>
</tr>
<tr>
<td>119</td>
<td>120 Dataforms/West</td>
<td>1517 W. Shaw Avenue</td>
</tr>
<tr>
<td>120</td>
<td>121 Zylka Design</td>
<td>5467 W Shaw Ave #103</td>
</tr>
<tr>
<td>121</td>
<td>122 United Parcel Service</td>
<td>P.O. Box 505820</td>
</tr>
<tr>
<td>122</td>
<td>123 Federal Express Corporation</td>
<td>P.O. Box 1140</td>
</tr>
</tbody>
</table>

Foreign key

<table>
<thead>
<tr>
<th>INVOICE_ID</th>
<th>VENDOR_ID</th>
<th>INVOICE_NUMBER</th>
<th>INVOICE_DATE</th>
<th>INVOICE_TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>29</td>
<td>123 4-314-3057</td>
<td>02-MAY-14</td>
<td>13.75</td>
</tr>
<tr>
<td>30</td>
<td>30</td>
<td>94 200339-13</td>
<td>02-MAY-14</td>
<td>17.5</td>
</tr>
<tr>
<td>31</td>
<td>31</td>
<td>1232-000-2993</td>
<td>03-MAY-14</td>
<td>144.7</td>
</tr>
<tr>
<td>32</td>
<td>32</td>
<td>89 125520-1</td>
<td>05-MAY-14</td>
<td>95</td>
</tr>
<tr>
<td>33</td>
<td>33</td>
<td>123 1-202-2978</td>
<td>06-MAY-14</td>
<td>33</td>
</tr>
<tr>
<td>34</td>
<td>34</td>
<td>110 0-2436</td>
<td>07-MAY-14</td>
<td>10976.06</td>
</tr>
<tr>
<td>35</td>
<td>35</td>
<td>123 1-200-5164</td>
<td>07-MAY-14</td>
<td>63.4</td>
</tr>
<tr>
<td>36</td>
<td>36</td>
<td>110 0-2060</td>
<td>08-MAY-14</td>
<td>23517.58</td>
</tr>
<tr>
<td>37</td>
<td>37</td>
<td>110 0-2058</td>
<td>08-MAY-14</td>
<td>37966.19</td>
</tr>
<tr>
<td>38</td>
<td>38</td>
<td>123 963253272</td>
<td>09-MAY-14</td>
<td>61.5</td>
</tr>
</tbody>
</table>
Relationships between tables

- One-to-many relationship
- One-to-one relationship
- Many-to-many relationship
Principle of data independence

- Physical data independence
- Logical data independence

Examples:
- Adding / dropping a column
- Adding / dropping an index
SQL Introduction

Standard language for querying and manipulating data

Structured Query Language

Many standards out there:
• ANSI SQL
• SQL92 (a.k.a. SQL2)
• SQL99 (a.k.a. SQL3)
• Vendors support various subsets of these
• What we discuss is common to all of them
Data Manipulation Language (DML) statements

- SELECT
- INSERT
- UPDATE
- DELETE

Data Definition Language (DDL) statements

- CREATE USER, TABLE, SEQUENCE, INDEX
- ALTER USER, TABLE, SEQUENCE, INDEX
- DROP USER, TABLE, SEQUENCE, INDEX
- GRANT
- REVOKE
The simplified syntax of the SELECT statement

```
SELECT select_list
FROM table_source
[WHERE search_condition]
[ORDER BY order_by_list]
```

The four clauses of the SELECT statement

- SELECT
- FROM
- WHERE
- ORDER BY
A simple SELECT statement

SELECT *
FROM invoices

(114 rows selected)
A SELECT statement that projects and sorts

```
SELECT invoice_number, invoice_date, invoice_total
FROM invoices
ORDER BY invoice_total
```

(114 rows selected)
A SELECT statement with no duplicate rows

```
SELECT DISTINCT vendor_city, vendor_state
FROM vendors
ORDER BY vendor_city
```

(53 rows selected)
A SELECT statement that retrieves a calculated value

```
SELECT invoice_id, invoice_total, 
       (credit_total + payment_total) AS total_credits 
FROM invoices 
WHERE invoice_id = 17
```