Outline

1. What is a Query? Query Language?
2. Example Database Tables
3. SQL Overview: 3 Components
4. SELECT statement with 1 table
5. Multi-table SELECT statements
6. Why spatial extensions are needed?
7. 1-table spatial queries
8. Multi-table spatial queries
9. Trends
Learning Objectives

• After this segment, students will be able to
  • Recognize simple spatial data types and operations
  • Recognize concepts from OGIS simple features library
  • Determine output of a simple SQL/OGIS query (without spatial join)
Extending SQL for Spatial Data: New

- SQL 3 allows user-defined data types and operations
  - Spatial data types and operations can be added
- Open Geodata Interchange Standard (OGIS)
  - Half a dozen spatial data types
  - Over a dozen spatial operations
  - Supported by major vendors, e.g. ESRI, Intergraph, Oracle, IBM,...
OGIS Spatial Data Model

- **Base-class:**
  - Geometry

- **Four sub-classes:**
  - Point
  - Curve, e.g., LineString
  - Surface, e.g., Polygon
  - GeometryCollection
    - PointCollection, PolygonCollection, LineStringCollection
OGIS Spatial Data Model: Operations

• Three Categories of Operations
  • Apply to all geometry types
    • SpatialReference, Envelope, Export, IsSimple, Boundary
  • Predicates for Topological relationships
    • Equal, Disjoint, Intersect, Touch, Cross, Within, Contains
  • Spatial Data Analysis
    • Distance, Buffer, Union, Intersection, ConvexHull, SymDiff
Spatial Operations: Exercise

Which topological operator is needed to report rectangles with (0,0) as an inside point?

a) Cross  
b) Equal  
c) Contains  
d) Touch  
e) Within
Spatial Queries with SQL/OGIS: General Information

- SQL3 and OGIS are supported by many vendors
- Syntax **differs from** vendor to vendor
- Readers may need to alter SQL/OGIS queries given in slide to make them run on specific products
Where is OGIS used within SQL?

- SQL Data Definition Language
  - Spatial data-types for columns in CREATE TABLE
- SQL Data Manipulation Language
  - Spatial operations with SELECT, INSERT, ...

Scope of our discussion
- Use of OGIS operations with SELECT statement
- Via a set of examples
Simple SQL SELECT_FROM_WHERE Examples

- Spatial analysis operations
  - Unary operator: Area
  - Binary operator: Distance
- Next Video
  - Spatial-Join using Topological operations
  - Touch, Cross
  - Using both spatial analysis and topological operations
    - Buffer, overlap
Unary Spatial Operation Area()

**Query**: List the name, population, and *area* of each country listed in the Country table

```
SELECT  C.Name, C.Pop, Area(C.Shape) AS "Area"
FROM    Country C
```

Note: This query uses spatial operation, Area() in place of a column in SELECT clause.
**Query:** List the GDP and the distance of a country’s capital city to the “Equator” for all countries.

```sql
SELECT Co.GDP, Distance(Point(0, Ci.Shape.y), Ci.Shape) AS "Distance"
FROM Country Co, City Ci
WHERE Co.Name = Ci.Country
AND Ci.Capital = ‘Y’
```

<table>
<thead>
<tr>
<th>Co. Name</th>
<th>Co. GDP</th>
<th>Dist-to-Eq (in Km.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Havana</td>
<td>16.9</td>
<td>2562</td>
</tr>
<tr>
<td>Washington, D.C.</td>
<td>8003</td>
<td>4324</td>
</tr>
<tr>
<td>Brasilia</td>
<td>1004</td>
<td>1756</td>
</tr>
<tr>
<td>Ottawa</td>
<td>658</td>
<td>5005</td>
</tr>
<tr>
<td>Mexico City</td>
<td>694.3</td>
<td>2161</td>
</tr>
<tr>
<td>Buenos Aires</td>
<td>348.2</td>
<td>3854</td>
</tr>
</tbody>
</table>
Spatial Operations: Exercise

Which topological operator is needed to list rivers flowing through Argentina?

a) Cross  
b) Equal  
c) Contains  
d) Touch