Internet Fundamentals & Introduction to Web Technologies

Course: IT (044615)

Lecture: 8

Database Access Through the Web

Dr. Ramez Hajislam

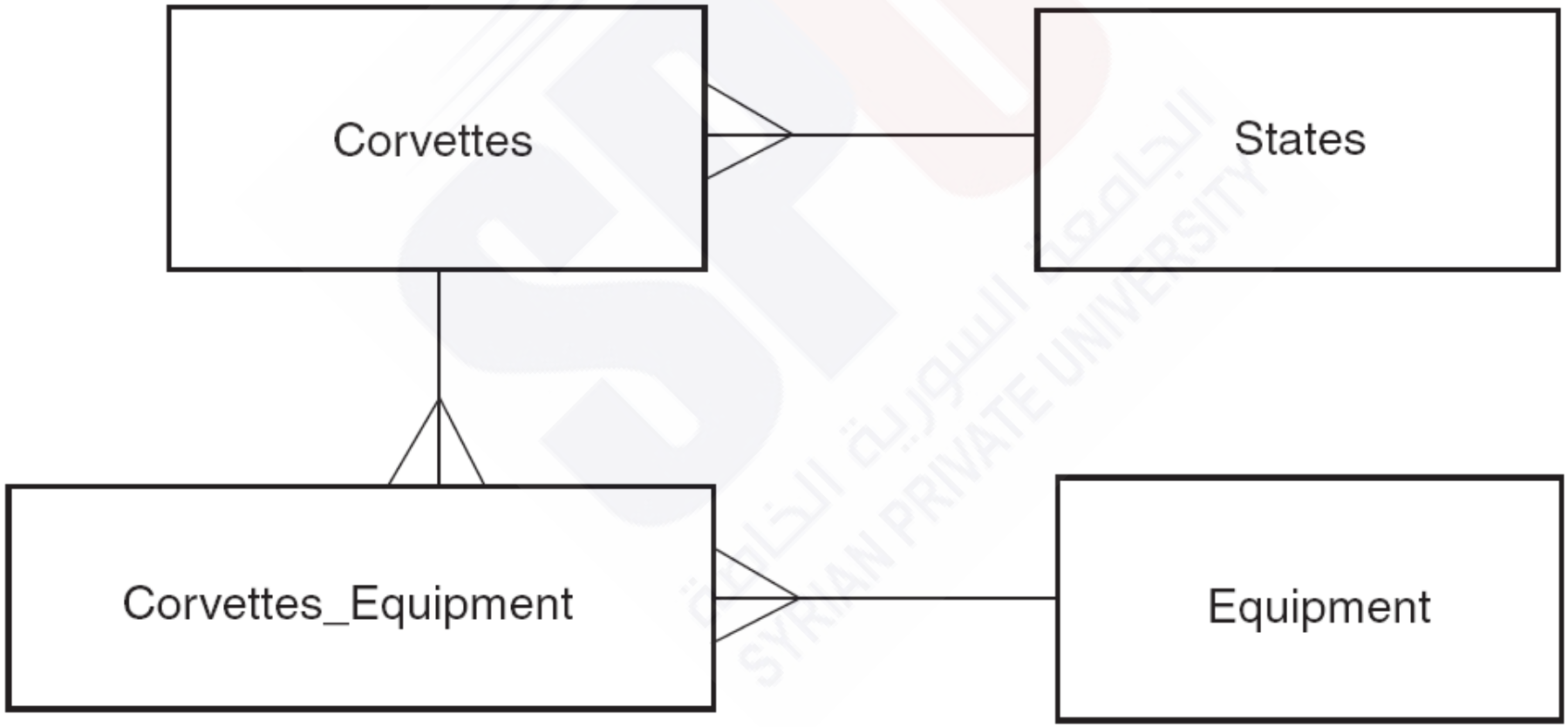
Relational Databases

- The relational model is currently the most popular model
  - Data is stored in tables
  - Columns are named
  - Each row contains values for each column, though some values may be missing
  - Rows are referred to as entities
  - The *primary key* is one or more columns in a table whose value(s) uniquely identify each row
- Example, Corvettes table
  - Primary key is an index number
  - Each row represents a different vehicle
  - Columns are important characteristics of the vehicles

Multi-valued Attributes

- Each state can, potentially, be associated with several cars
  - Each state could have important data, besides the name
  - A separate State table is created with an index primary key
  - Each entity in the Corvettes table refers to the state index
  - That way, changes in information about a state would not have to be repeated on each line of the Corvettes table
- Each type of equipment could appear in many cars, each car could have many types of equipment
  - A table describing equipment is set up
  - A table giving the Corvette to Equipment relation is set up
    - This just has pairs of id's: Corvette-id and Equipment-id

Logical Data Model for Corvettes DB



Structured Query Language

- SQL is a standardized language for manipulating and querying relational databases
- Although relational databases support SQL there may be some minor and some significant differences in the implementations
- SQL reserved words are not case sensitive
  - However, some systems may treat names such as column names as case sensitive
- SQL commands may have extra white space, including new lines, added to improve readability
- Single quotes ‘ are used for literal strings

The SELECT Command

- Used to query databases
- The command returns a result, a virtual table
- SELECT *column-names* FROM *table-names* [WHERE *condition*];
  - The result table has columns as named
  - Rows are derived from the table named (see the Join discussion about multiple tables)
  - The WHERE clause is optional
  - The WHERE clause specifies constraints on the rows selected
  - If * is used for the column names, all columns are selected

Joins

- Task: list corvettes that have CD players
- This involves three tables: Corvettes, Equipment, Corvettes_Equipment
- A virtual table is constructed with combinations of rows from the two tables Corvettes and Equipment: a *join* of the three tables
  - Or from all combinations of all three tables
- The WHERE clause selects which rows of the join are to be retained in the result

A Query Using a Join

```
SELECT Corvettes.Vette_id,
      Corvettes.Body_style,
      Corvettes.Miles, Corvettes.Year,
      Corvettes.State,
      Equipment.Equip
FROM Corvettes, Equipment
WHERE
      Corvettes.Vette_id =
                  Corvettes_Equipment.Vette_id
      AND Corvettes_Equipment.Equip =
                  Equipment.Equip_id
      AND Equipment.Equip = 'CD';
```

The INSERT Command

- Inserts a new row into a table
- Syntax

```
INSERT INTO table_name
(column_name_1, ..., column_name_n)
VALUES (value_1, value_2, ..., value_n);
```

- The values provided will be placed into the corresponding columns
- Columns not named will receive no value
  - This will cause an error if the column was created with a NOT NULL constraint

The UPDATE Command

- Changes values in an existing row
- Syntax

```
UPDATE table_name
SET column_name_1 = value_1,
    ...
    column_name_n = value_n
WHERE column_name = value
```

- The WHERE clause identifies the row to be updated, probably by its primary key

The DELETE Command

- Removes one or more rows
- Syntax

```
DELETE FROM table_name
WHERE column_name = value;
```

- The WHERE clause determines which rows are deleted
- The sample syntax would probably be specifying a primary key value to identify one row
- However, the clause could be more general

The DROP Command

- Remove a table or database from the system
  - A database system usually has several databases operating within it, essentially, named collections of tables
- Syntax

```
DROP (TABLE|DATABASE)[IF EXISTS] name;
```

- The IF EXISTS clause may be included to prevent an error indication if the table or database doesn't exist

The CREATE TABLE Command

- Create a table with specified columns, each column having a specified type of data and satisfying certain constraints
- Syntax

```
CREATE TABLEtable_name(
column_name_1 data_type constraints,
...
column_name_n data_type constraints);
```

- Most system support many data types
- Common types: INTEGER, REAL, DOUBLE, CHAR(*length)*

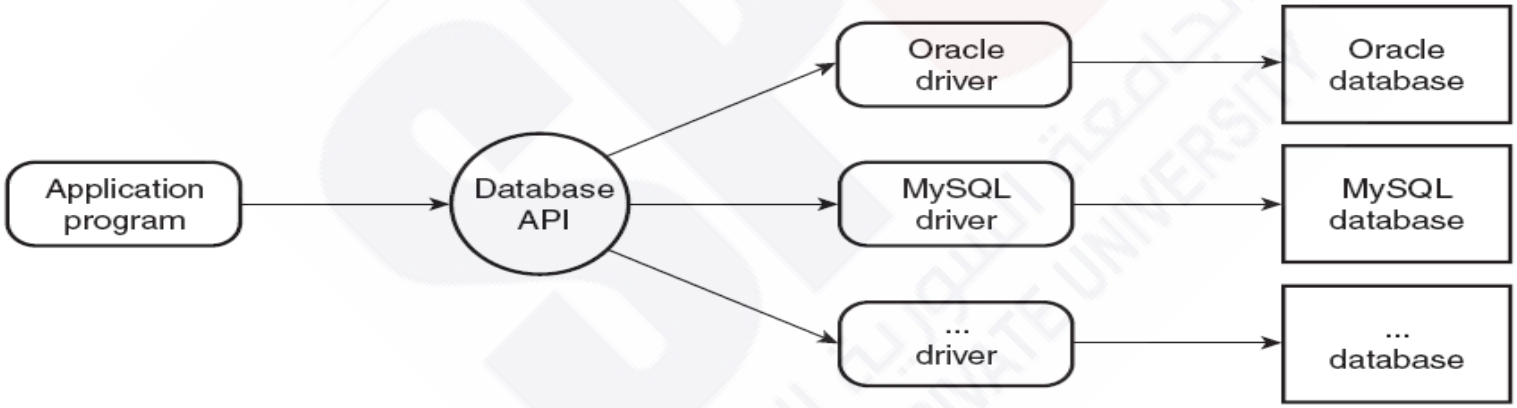
Create Table Constraints

- The constraint NOT NULL causes an error to be raised if a row is inserted in which the corresponding column does not have a value
- The PRIMARY KEY constraint causes an error to be raised if a row is inserted in which the corresponding column has a value that equals the value in another row
  - This can be applied to a group of several columns if the primary key is multi-column

Client/Server Database Architecture

- Two-tier architecture
  - Client connects to the database to get information
  - Server or client performs computations and user interactions
- Problems with two-tier
  - Servers getting smaller so client software getting more complex
  - Keeping clients up to date difficult
- Three-tier architecture
  - Web server with applications sits between a browser and the database system
  - The web server accesses the database and carries out computations and deals with user interaction

Database Access Architecture



PHP and Database Access

- There are modules available in PHP to access numerous different database systems:
  - MySQL
  - SQL Server
  - etc ..

The MySQL Database System

- Logging in to MySQL

```
mysql [-h host] [-u username]
      [database_name] [-p]
```

  - Starts an interactive shell sending commands the server
  - host indicates the MySQL server host, defaults to local host
  - username, if absent, defaults to login user name of the current user
  - database_name, if present, selects a database for commands
  - -p indicates a password is needed
- Connecting to a database is necessary
  - Either on the initial command line
  - `use database_name`

MySQL Commands

- MySQL supports a large subset of standard SQL
- Other commands
  - `CREATE DATABSE database_name;`
  - `SHOW TABLES;`
  - `DESCRIBE table_name;`

Connecting to MySQL

- The `mysql_connect` function
  - First parameter is MySQL server host
  - Second parameter is the MySQL username
  - Third parameter is the password
  - Returns false if it fails
- The `mysql_close` function
- Selecting a database with `mysql_select`

Requesting MySQL Operations

- The `mysql_query` function
  - Takes a string parameter with an SQL query
  - Returns a result object
- Functions that apply to the result object
  - `mysql_num_rows` returns number of rows in result
  - `mysql_num_fields` returns the number of fields (columns) in the result
  - `mysql_fetch_array` returns an array with the next row of results
- Each array with a row from the result contains each field value indexed by position and by column name
  - The `array_values` applied to this array has each value twice, once for each possible index

PHP/MySQL Example

- The example with `carsdata.html` and `access_cars.php` allows users to submit SQL commands that are executed against the Corvette database
- The two files could be combined, `access_cars2php`
  - A hidden text field is used to hold a value that tells the script whether the script whether this is an initial request for the page or a second request with values from the form