# Objectives

- Differentiate between a query and a table
- Use design grid to create and modify a select query
- Explain multiple criteria rows in a query
- Look at multiple-table queries
- Learn and use SQL to write your own queries

# Introduction to Queries

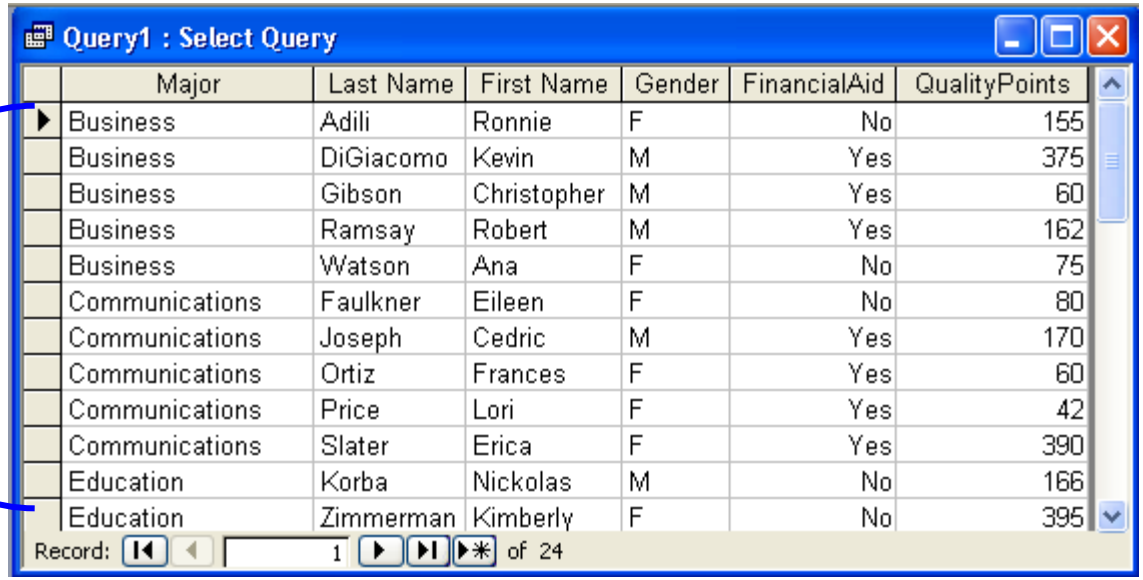
- Query provides the resulting records from a question
- Design grid: graphical tool used to develop queries
- Dynaset: The records that satisfy the query criteria

# Dynaset

- Looks and acts like a table
- Is a dynamic subset of a table
- Allows for entry, updating and deletion of records
- Changes update the underlying table

# Select Query Dynaset

The results of a query are displayed as a dynaset



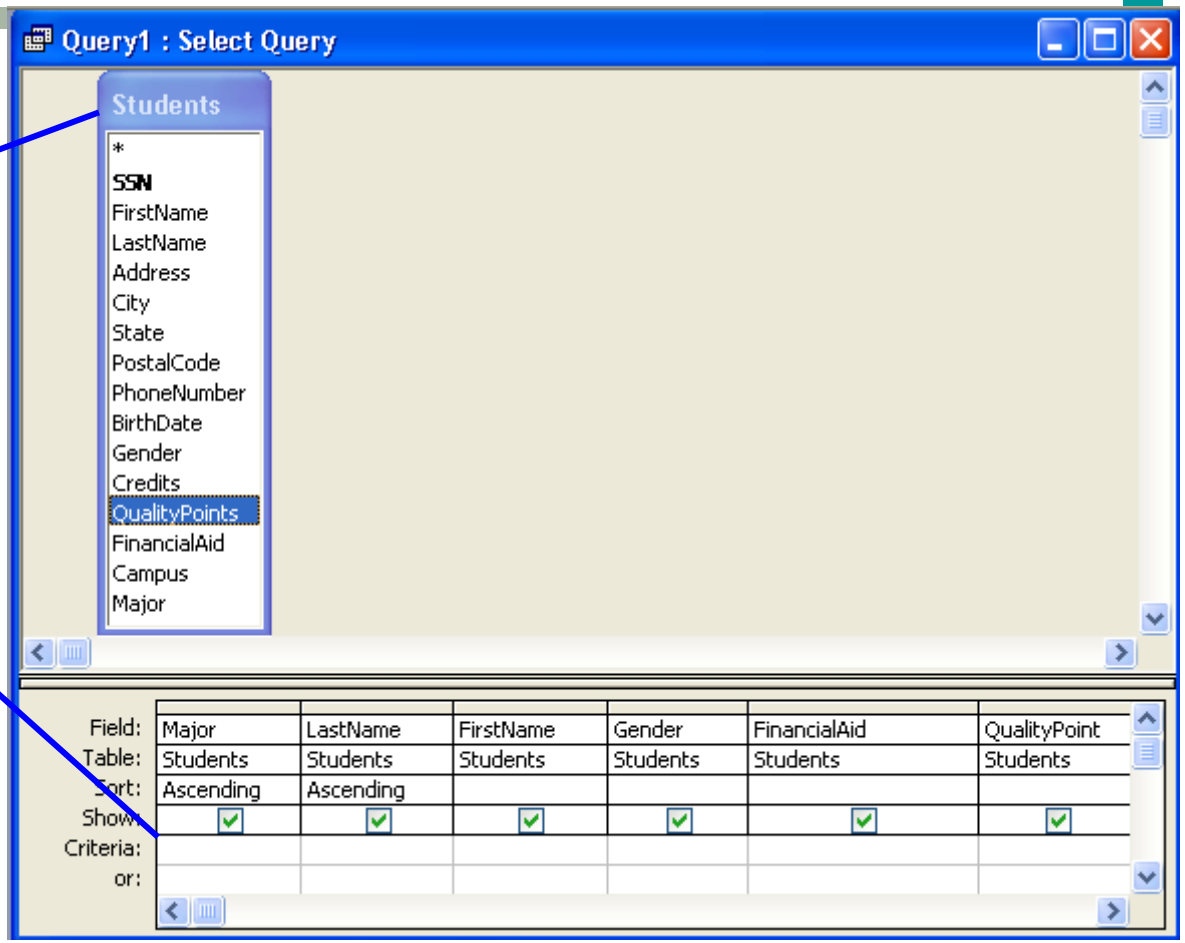
# Creating a Query

- Select query
- Query window views
  - Datasheet view
  - Design view

# Select Query Design View

Table

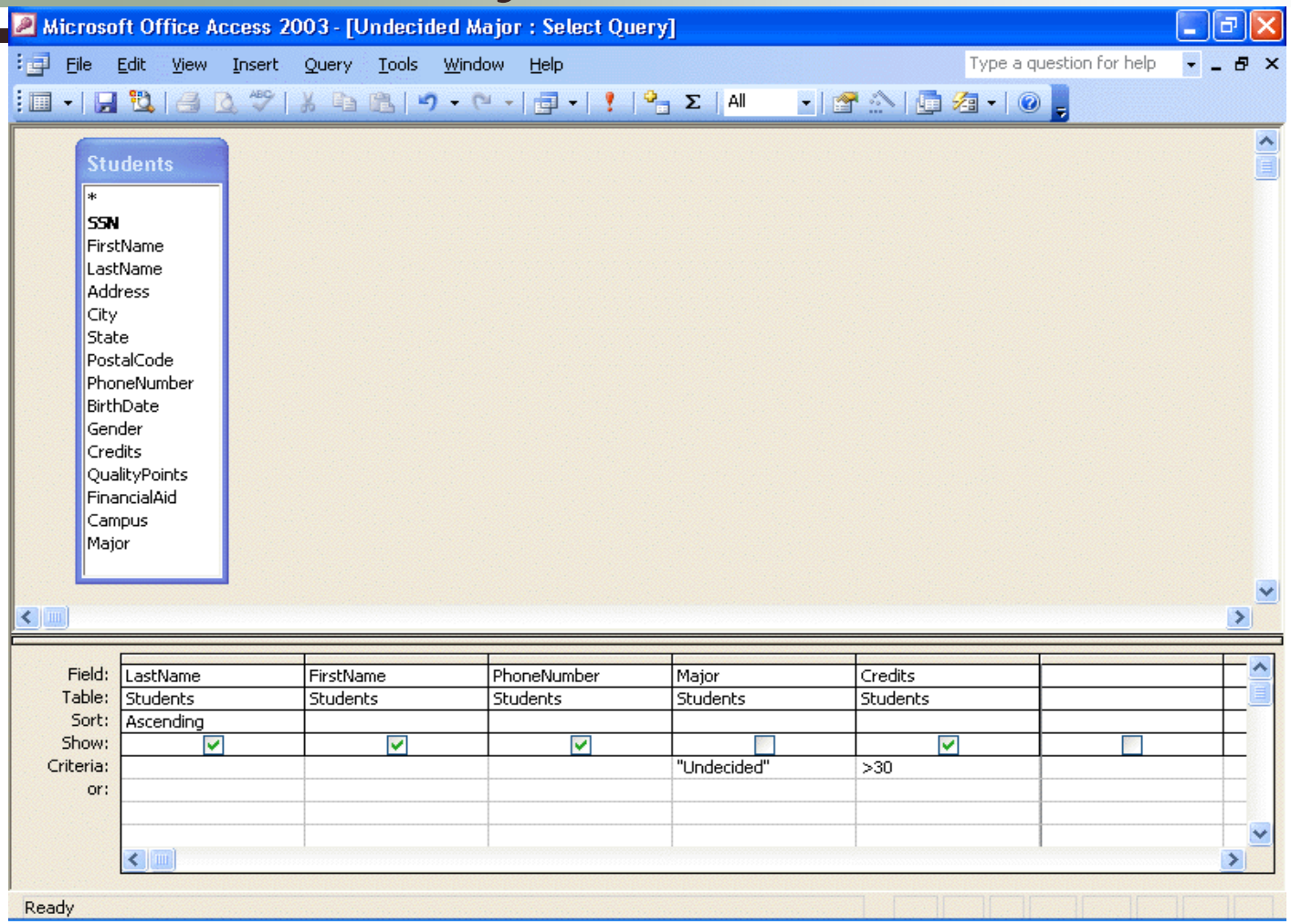
Criteria



# Query Selection Criteria

- AND condition
- OR condition
- Relational operators

# “And” Query


Microsoft Office Access 2003 - [Undecided Major : Select Query]
File Edit View Insert Query Tools Window Help
Type a question for help
All
Students
*
SSN
FirstName
LastName
Address
City
State
PostalCode
PhoneNumber
BirthDate
Gender
Credits
QualityPoints
FinancialAid
Campus
Major
Field: LastName FirstName PhoneNumber Major Credits
Table: Students Students Students Students Students
Sort: Ascending
Show:
Criteria: "Undecided" >30
or:
Ready

# Multiple Table Queries

- Query information from two tables
- Create a select query using both tables
- Query window displays selected fields of each table

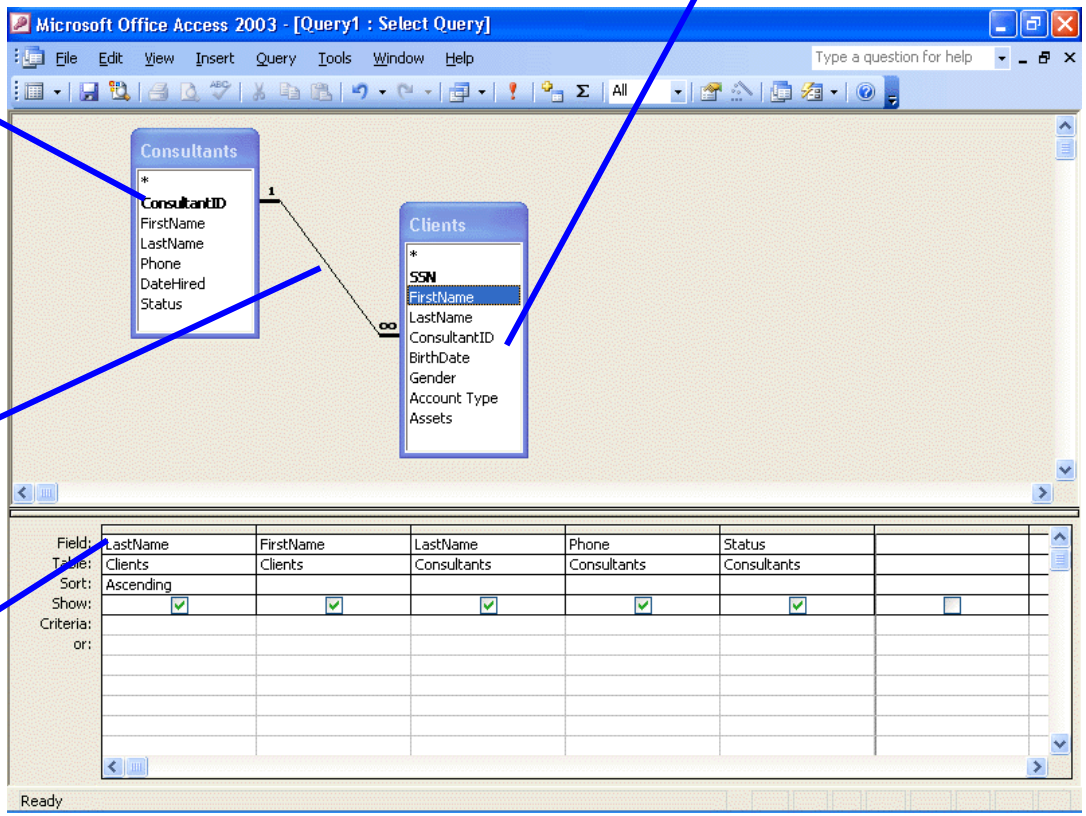
# Multiple Table Query

Primary key in Consultants table

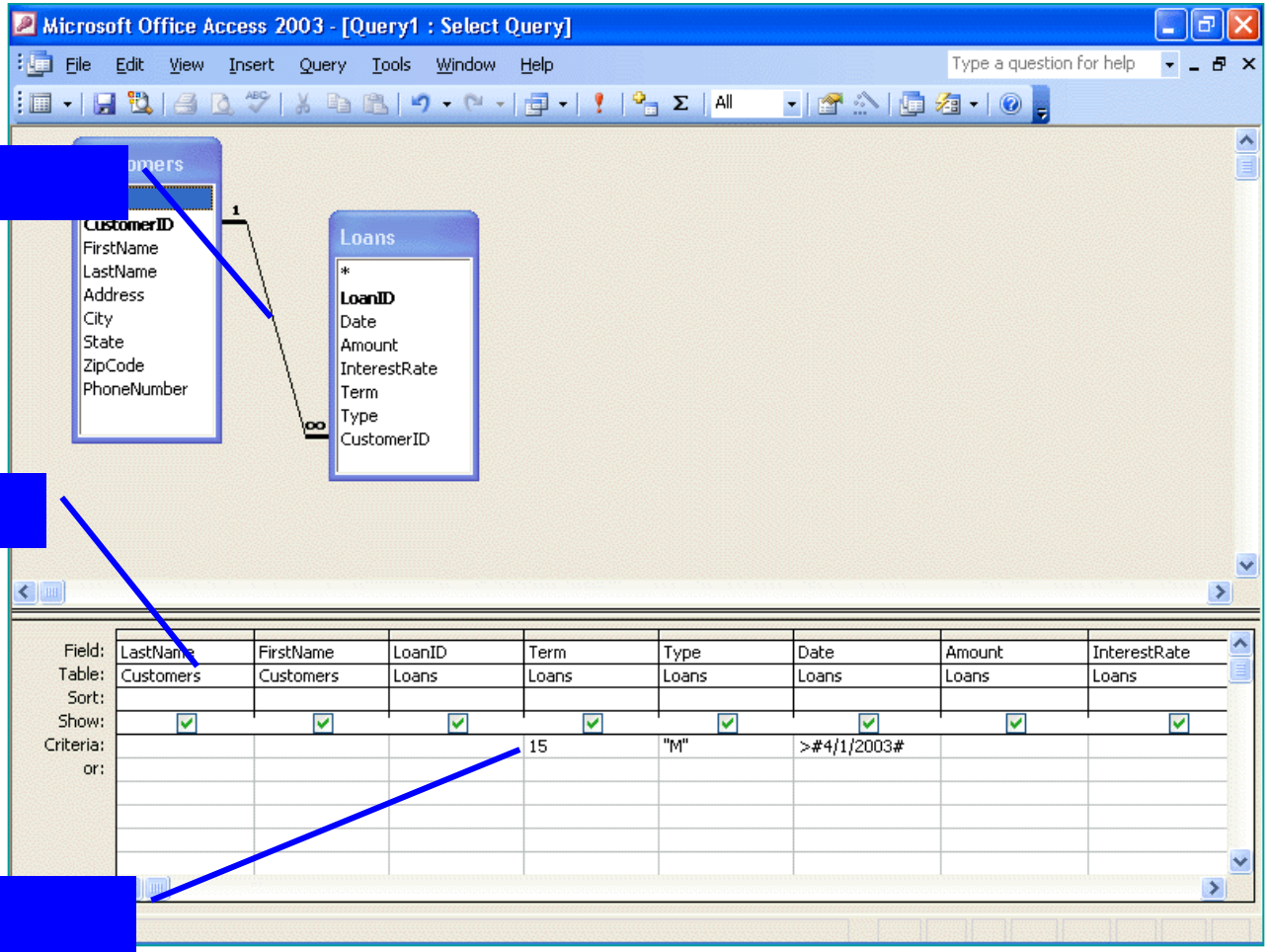
Relationship between tables (one to many)

Each field and table to display

Foreign key in Clients table



# Multiple Table Queries




Join line

Table row

Criteria for query

# What is SQL?

- SQL - Structured Query Language
- SQL is used to communicate with a database
  - it is the standard language for relational database management systems
- SQL statements are used to retrieve data from a database.
- most database systems use SQL

# SELECT, FROM, WHERE

- The field names that follow the **SELECT** keyword determine which fields will be returned in the results.
- The table name that follows the keyword **FROM** specifies the table that will be queried.
- The **WHERE** clause (optional) specifies which data values or rows will be returned or displayed.

# Selecting Data

- The **SELECT** statement is used to query the database
  - retrieve selected data that match the criteria that you specify
- Here is the format of a simple select statement:

```
SELECT field1 [,field2,etc]
FROM table
[WHERE condition];
```

- The [] denotes optional

# Conditional Operators

- Conditional operators used in the **where** clause:

| | |
|---|---|
| = | Equal |
| > | Greater than |
| < | Less than |
| >= | Greater than or equal |
| <= | Less than or equal |
| <> | Not equal to |

# Example

```
SELECT name, age, salary
FROM employee
WHERE age > 50;
```

- Here we select the name, age and salary of all the employees who are over 50.
- **Note:** Remember to put a semicolon at the end of your SQL statements.
  - The **;** indicates that your SQL statement is complete and is ready to be interpreted.

# Boolean Operators (AND)

- The **AND** operator can be used to join two or more conditions in the **WHERE** clause.
- Both sides of the **AND** condition must be true in order for the condition to be met.

```
SELECT list-of-fields
FROM list-of-tables
WHERE condition1 AND condition2;
```

# Boolean Operators (OR)

- The **OR** operator can be used to join two or more conditions in the **WHERE** clause also.
- However, **either** side of the **OR** operator can be true and the condition will be met
  - With the OR operator, either side can be true or both sides can be true

# Examples

```
SELECT firstname, lastname, title, salary
FROM employee_info
WHERE salary >= 50000 AND title =
   “Programmer”;

SELECT firstname, lastname, title, salary
FROM employee_info
WHERE (title = “Sales”) OR (title =
   “Programmer”);
```

# Multiple Table Queries

- "Joins" allow you to link data from two or more tables together into a single query result--from one single SELECT statement.
- A "Join" can be recognized in a SQL SELECT statement if it has more than one table after the FROM keyword.
- For example:

```
SELECT list-of-fields
FROM table1,table2
WHERE search-condition(s)
```

# Example of a Join

```
SELECT customer_info.firstname,
    customer_info.lastname,
    purchases.item
FROM customer_info, purchases
WHERE
    customer_info.customer_number =
    purchases.customer_number;
```

- Note that you must distinguish which table a field comes from.