Basics of Relational Databases

You want to organize your data such that you don't have duplicate information in your database unless absolutely necessary. You do this to maintain your *data integrity* – if need to change the data, you want to do it in exactly one place so that the information stays correct.

Tables

A *table* is a collection of data about a specific topic, such as products or suppliers. Using a separate table means you store that data only once, which makes your database more efficient and reduces data-entry errors. Tables organize data into columns (called *fields*) and rows (called *records*).

Queries

You use queries to view, change, and analyze data in different ways. You also use them as the source of records for forms and reports. There are these types of queries:

- Append Query
- Select Query
- Cross-tab Query
- Make-table Query
- Update Query
- Delete Query

Index/Primary Key

An *index* is placed on one or more fields of a table when those fields are going to be used to look up data in the table. Indexes can be *unique* (the entries in this column(s) cannot be duplicated) or *primary* (described below). Indexes help to search through tables more effectively.

The *primary key* of a table is a special type of index. This index is one or more fields whose value or values uniquely identify the record in the table. In a relationship, a primary key is used to reference a specific record in one table from another table. A primary key is called a *foreign key* when it is referred to from another table.

Referential Integrity

Referential integrity is a system of rules that Microsoft Access uses to ensure that relationships between records in related tables are valid, and that you don't accidentally delete or change related data. You can set referential integrity when all of the following conditions are met:

- The matching field from the primary table is a primary key or has a unique index.
- The related fields have the same data type. There are two exceptions. An AutoNumber field can be related to a Number field with a FieldSize property setting of Long Integer, and an AutoNumber field with a FieldSize property setting of Replication ID can be related to a Number field with a FieldSize property setting of Replication ID.
- Both tables belong to the same Microsoft Access database. If the tables are linked tables, they must be tables in Microsoft Access format, and you must open the database in which they are stored to set referential integrity. Referential integrity can't be enforced for linked tables from databases in other formats.

When referential integrity is enforced, you must observe the following rules:

- You can't enter a value in the foreign key field of the related table that doesn't exist in the primary key of the primary table. However, you can enter a Null value in the foreign key, specifying that the records are unrelated. For example, you can't have an order that is assigned to a customer that doesn't exist, but you can have an order that is assigned to no one by entering a Null value in the CustomerID field.
- You can't delete a record from a primary table if matching records exist in a related table. For example, you can't delete an employee record from the Employees table if there are orders assigned to the employee in the Orders table.

- You can't change a primary key value in the primary table, if that record has related records. For example, you can't change an employee's ID in the Employees table if there are orders assigned to that employee in the Orders table.

If you want Microsoft Access to enforce these rules for a relationship, select the Enforce Referential Integrity check box when you create the relationship. If referential integrity is enforced and you break one of the rules with related tables, Microsoft Access displays a message and doesn't allow the change.

File Format

Microsoft Access stores all of the pieces of the database in a `.mdb` file. This is the file that you access from Visual Basic or Visual C++ to get at the tables. *Note*: This is all in ONE file so be careful when you have multiple group members working changing the *structure/design* of the database. The database is supposed to be able to support multiple *users* hitting the database at once, but multiple people changing the design of the database is likely to make it unhappy. I am not sure how gracefully it handles this – it may just write the changes the last person made without letting you know that someone else made changes.

References

The best guide for the information that you will need for this project will be found in the Access on-line help. You may have to search around a little, but their examples are useful. Access comes with a sample database (the Northwind database) that you can look at to see how they put some stuff together. You won't be using native Access forms, but you may use Access's reporting features.