Today’s workshop is to give you an introduction to databases and to Microsoft Access.

The biggest goal of this workshop is to teach you to PLAN your database. And from that plan we will use the Tables, Queries, Forms and Reports to build our Access database.

Before you begin your database you should contact your Customer Support. You want to see if they can help you with questions as they arise, if needed setting up secure folders and how difficult it will be to recover a file from their network backup. To be safe you should be doing regular backups of your database.

It’s so important I have it up here three times, backup, backup, backup. Access will eventually crash on you. Sometimes it’s not a big deal, but it’s always better safe than sorry.

Some definitions -- what is a database? It’s a structured collection of related data. In its simplest form we can have a filing cabinet or a phone book. An electronic database is one we can save, retrieve, and sort data.

In Access, the true Database is the collection of tables. That’s where the data is stored, the base of data.

We hear the words Data and Information used interchangeably, but we need to differentiate them. Data is the raw facts, the raw Data. Text, numbers, dates.

Information is giving meaning to that raw data. Arranging the data so that anyone who sees it should understand its meaning.

We are creating a Database, thus our tables will be made up of the raw data. We can use the other tools (forms/reports) to display information.
When I look at this slide I like to think of a rolodex. This is Sally’s rolodex; all of her contacts kept here make up a TABLE of contacts. If I pull out one card from the rolodex I will have one RECORD. The record contains all of the data about one individual contact. Each card in the rolodex has a blank for a Name, that’s called a field; a single item of data common to all the records.

A collection of fields make up a record. A collection of records make up a table. A collection of tables make up a database.

If we were to take the rolodex and turn it into an electronic format it would look more like this, what we commonly think of as a “table”.

Here each row is a record, each column a field.

Let’s say I have a second phone number for Robin Smith. In Word or Excel, I might be tempted to add a new row for the second number. But in Access we should create a new field, a new column, so that every record in our table can have a second phone number.

As it says here, a designer’s best tools are paper and pencil. Planning is an essential part of building your database.

The sooner you touch the computer the sooner you’ll make a mistake. Many things can be changed after the database is created, but if there is an underlying logic flaw, the whole thing may need to be recreated.

If you have the time, it’s a very good idea to document what you are doing. Notice your packet has several images from Access. Those were obtained with the Print Screen button on the keyboard.

Here are some general questions to help you with your Planning. What have I got? What do I want? What do I need to get there? and How am I going to build it. This is never a linear process.

Example:

**What have I got?** The sign in sheet that shows the time a patient arrives and the time of their appointment.

**What do I want?** To see how long patients are waiting.

**What do I need to get there?** We need to change the process of the office so that this information is collected as well. Or go back and revisit what information actually needs to be output.

**How am I going to build it?** If the forms will be filled out by hand, sometimes it's easier to create them in Word or Excel. Excel can also very easily be imported to Microsoft Access.

We make charts of our data for the staff meeting and notice this doesn’t tell me how long a patient was waiting; it tells me how early they got here for their scheduled appointment. There are lots of different times: Time they arrived, time of appointment, and time the doctor sees them.
There are many options to help you pull together your database. If you are going to build a simple one like our rolodex example you can find inexpensive, possibly free copies of rolodex programs. You can even get cute ones that make little flipping noises as you search through the 'cards'.

If it’s a simple set of data you may decide to use Microsoft Excel. Excel has many database functions, and is often better at creating charts and analyzing data than Access. But it's essentially a one person at a time, one dataset at a time, program. Excel also has a limited number of rows.

If you are working with multiple sets of data that you need to interweave to create your reports, or you want to make nice data entry forms for complex data entry, you want to move into Access. And unlike the Shareware or Excel files, Access can let more than one person at a time access the data. Keep in mind though, the more people you have playing in the database the more likely it will crash. So if you want multiple people to work on the database, make sure you are doing your backups.

If you need a huge number of people to be able to use your database, or you want to put data securely out on the internet you have moved beyond the scope of Access. You probably want to look at creating an Oracle or SQL database. For that you would need to hire a programmer.

With all the choices out there why would you use Microsoft Access?

- It has the familiar look and feel of Windows. We have a lot of the same buttons and shortcut keys as Word and Excel.
- It is easy to build simple databases; we’ll do one together today. But you can build very sophisticated systems.
- It should already be on your computer, everything we do today, you should be able to do at your desk. If not, contact your customer support.
- The power Access has over Excel is you can create a relational database.

A relational database is a collection of tables from which the data can be accessed in many ways without reorganizing the tables. We can “link” different sets of data to pull out the information we want.

For example, let’s say I’m keeping track of upcoming scheduled appointments in Excel and that I have a spreadsheet of patients with all their demographic information.

Dr. Smith says, “I need to cancel next week's appointments.” So in Excel we filter the appointments for Dr. Smith, within the next week. We print that list and one by one look them up in our Patient spreadsheet.

If my data was in Access, I could link the patients with the appointments, and using a simple query I can show Dr. Smith patient's data for next week in the same time it would take to create the filter in Excel.
There are a few design rules we’d like you to keep in mind as you are building your databases. Remember in Access the true database is the collection of tables; apply these rules when you are designing your tables.

**Rule 1. Organizing Data.** In Excel it often makes sense to add columns for each occurrence. Each semester, each visit, each evaluation. But because Access is a relational database, we're able to put these into another table. This makes searches easier since we'll no longer have to search each occurrence column, and it gives us an unlimited number of possibilities.

**Rule 2. No Derived Fields.** A derived field would be one that's found in another table. In Excel we might be tempted to repeat information. Instead of having to look up a patient's phone number every time we need to talk to them, we can include it in their appointment. But what if their number changes? Then we have to remember to change it EVERYWHERE. By keeping that value in the database just one time (with the patient) we avoid mismatched data. We can pull the data from multiple tables into a query, form, or report. The only field that should be in both tables is an identifier to help connect, relate, the tables.

**Rule 3. Smallest Logical Parts.** You want to make sure your data is broken down into the smallest logical parts, I like to think of it as the smallest “sortable” parts. By keeping data points in different fields we have more flexibility. If the name is separated we can run reports for "First Last" and "Last, First".

It’s fairly simple to pull the data back together, but it’s very difficult to pull it apart if it’s saved this way. If you will need to sort or filter the data points, try to keep them in their own field.

**Rule 4. Descriptive Field Names.** When considering your field names, try not to use too many abbreviations. You are allowed to have 64 characters. When you view a datasheet view of a table, query, or form, the field names (column headings) will not wrap, so you also want to be concise.

Be Clear, Be Concise, and Be Consistent.

**Rule 5. Unique Field Names.** Try to make your fieldnames as clear as possible in each table and keep in mind the fieldnames you’ve used in other tables. Access won’t let you use the same fieldname twice in one table; however it won’t look at the other tables.

There is no “harm” in using the same names, but it may cause confusion as you relate them together. Access will try to compensate by putting the table name in front of the fieldname. But this may cause problems if you decide to add data to already existing queries, forms, or reports.
Basic Design Rules

- **No Calculated Field.**
  In Microsoft Excel we enter the data and create our formulas all at once. In Access, you are creating a “Data” table, a table of the raw data. If you want Access to do the calculations, you can create an expression elsewhere in the database.

  ![Table](image)

  You can create calculated expressions in queries, forms, and reports.

Rule 6. No Calculated. This is the rule that Excel users have the most trouble with. In Excel we put the data and calculations in the same place. But in Access, your tables are created to store the “raw” data. The data lives in the tables, the calculations are done in the queries, forms, and reports. You can’t do calculations in a table, and if you put the “answer” field in the table, you have to constantly update the answer when you update the data.

  *New to Access 2010, you can create a calculated field in the tables.*

Basic Design Rules

- **Unique Records.**
  If you don’t have unique records, your database can’t tell which record you may be referring to.

  ![Table](image)

  To ensure that each record is unique in each table, we can set one field to be a Primary Key field.

  A Primary Key is a field that will contain no duplicates and no blank values.

  Looking at the table above, what would be the best Primary Key?

Rule 7. Unique Records. Try to keep unique records in your data tables.

For example, if we link this table of employees with a table of Emergency Contacts, Access won’t know which “Smith” to use. Mary Anne Smith will show up for each of the Smiths in our employee table anywhere we include both tables.

Primary Keys

![Table](image)

To ensure uniqueness, we create a **Primary Key** field. A primary key is a field (column) that contains no duplicates and no blanks (null values).

When you set a field to be the Key, access will check every entry to make sure it fits these rules: no blanks, no duplicates.

The table that’s shown here has a set of employee information, which field would make the best key?

Access will let us set any one of these columns as the primary key, because none of them have blanks or duplicates. However, we have to think about future entries as well. It’s very likely to have two employees with the same name, same phone numbers and multiple people from each college. For us the best key is going to be the GatorLink ID, because it’s supposed to be a unique identifier.

Many times there isn’t an apparent key in your data set, or it may not be feasible. For example, some of our Shands Jax students don’t have GatorLink IDs.

It is always safe to let Access add its own key, an AutoNumber. AutoNumbers will give each record a unique number that can be used as our key.
Basic Design Rules

- Unique Records
  
  We use the unique primary key as our link between our tables, this helps ensure we connect to the correct record.

<table>
<thead>
<tr>
<th>ID</th>
<th>Lastname</th>
<th>GatorLink</th>
<th>Phone</th>
<th>College</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Smith</td>
<td>rsmith</td>
<td>3-5051</td>
<td>Pharmacy</td>
</tr>
<tr>
<td>3</td>
<td>Smith</td>
<td>rsmith</td>
<td>273-5651</td>
<td>COP</td>
</tr>
<tr>
<td>4</td>
<td>Thomas</td>
<td>bthomas</td>
<td>392-5555</td>
<td>Medicine</td>
</tr>
<tr>
<td>5</td>
<td>VanWinkle</td>
<td>sleepyguy</td>
<td>846-5656</td>
<td>PHHP</td>
</tr>
</tbody>
</table>

We use the unique primary key as our link. So Smith doesn’t have an emergency contact of Mary Anne, employee number two does.

Better yet once these are related, our data is protected. If I try to delete Smith, employee 1, the record is deleted without question. If I try to delete Smith, employee 2, Access will give me an error message saying it can’t be deleted because there is related information in table Emergency Contacts.

Most Tables are set up with AutoNumbers for keys. There is a utility query called "FIND DUPLICATES" to help you keep your data clean.

Let’s Start Planning

We are going to make a simple Patient Appointment Database. First let’s think about what things we would want to keep track of about our patients. This is just a general “what have I got”, information gathering.

<table>
<thead>
<tr>
<th>Name</th>
<th>Date of Last Visit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
<td>DOB</td>
</tr>
<tr>
<td>Phone</td>
<td>Medications</td>
</tr>
<tr>
<td>Age</td>
<td>SSN</td>
</tr>
<tr>
<td>Medical Record #</td>
<td>Insurance</td>
</tr>
<tr>
<td>Emergency Contact</td>
<td>Allergies</td>
</tr>
<tr>
<td>Height</td>
<td>...</td>
</tr>
<tr>
<td>Weight</td>
<td></td>
</tr>
</tbody>
</table>

Rule 1 – Organizing Data: First we gather all the information thing we need to do now that we have our information gathered is should all of this really be in one table? Let’s look at the “medications”. If we create a field for medications, it leaves it very open ended. Will we spell Xanax the same every time? Or will we use the generic name occasionally?

One thing we can do is make separate table of just medications. In our Patient Table we can then create a drop down list to help us pull in the correct data. But if we do this we only get one medication per drop down list. So we will need to make a column for each medication. How many medications are you going to limit a patient to?

<table>
<thead>
<tr>
<th>Pt Name</th>
<th>Pt Med 1</th>
<th>Pt Med 2</th>
<th>...</th>
<th>Pt Med 50</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ann</td>
<td>xyz</td>
<td>xxz</td>
<td>...</td>
<td></td>
</tr>
<tr>
<td>Bob</td>
<td>xxz</td>
<td>xyz</td>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

Keep in mind there are only 255 fields per table; this will fill up quickly. It also causes problems when we are trying to find every one taking a particular medication, we would need to check every medication field.

There is also a problem if we want to know more than just the name of the medication. What about the dosage, the directions, who prescribed, why it was prescribed and so forth. If we decide to keep track of this for every single medication then we end up with too many columns.

The solution is to keep track of the medications, separately from the patients themselves.
We end up with three tables to handle our medications: Patients, Medications, and Patient Medications. This Patient Medication table will let you look up your Patients, look up your Medications and fill in the rest of the information about that particular medication (prescription) for that particular patient.

This way we can pull all the patients taking medication X,

<table>
<thead>
<tr>
<th>Pt</th>
<th>Med</th>
<th>Dosage</th>
<th>…</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>X</td>
<td>10 mg</td>
<td>…</td>
</tr>
<tr>
<td>B</td>
<td>Y</td>
<td>5 mg</td>
<td>…</td>
</tr>
<tr>
<td>B</td>
<td>Z</td>
<td>5 mcg</td>
<td>…</td>
</tr>
<tr>
<td>C</td>
<td>X</td>
<td>10 mg</td>
<td>…</td>
</tr>
<tr>
<td>C</td>
<td>Y</td>
<td>5 mg</td>
<td>…</td>
</tr>
</tbody>
</table>

or all the medications taken by Patient C.

<table>
<thead>
<tr>
<th>Pt</th>
<th>Med</th>
<th>Dosage</th>
<th>…</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>X</td>
<td>10 mg</td>
<td>…</td>
</tr>
<tr>
<td>C</td>
<td>X</td>
<td>10 mg</td>
<td>…</td>
</tr>
</tbody>
</table>

So let’s look at our list of Patient Info again:

- Medications definitely go into a new table.
- What about Insurances? We can have more than one of those and need more information than just the name.
- Emergency Contact can be its own table as we could allow each patient to have various contacts. Some may have none, some may have many.
- We don’t want to put a limit on how many Allergies a person can have either and we may want to keep track of their reactions, so this would work better in another table.
- Measurements like Height, Weight, Blood Pressure, and Temperature can vary from one visit to the next. If we want to keep a history of these measurements, it’s better to record them with the individual visit.

Rule 2 – No Derived fields. For the dataset we have chosen, we are ok. But an example of a derived field into the patient's table would be the doctor's name, Medicaid ID Number and phone number. All of those could be kept in a Doctor table.

Rule 3 - Smallest Logical Parts.
  a. Name needs to be broken down into at least first name and last name.
  b. Address should probably be broken into Street Address, City, State and Zip
  c. Should Phone be broken up into Home Phone/Work Phone/Cell Phone? Or can we just call this the Primary Phone Number
Rule 4 – Descriptive Field Names
a. Do our acronyms make sense? Does everyone in the department know that DOB is Date of Birth not Department of Bread? Should we change it to Birth Date to avoid confusion?

Rule 5 - Unique Field Names
a. What about across the tables? Will First Name appear anywhere else in our database? What if we make a Doctor table? To fix this we can put a PT in front of each of our field names.
⇒ Remember that field named can be up to 64 characters.

<table>
<thead>
<tr>
<th>First Name</th>
<th>Pt First Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last Name</td>
<td>Pt Last Name</td>
</tr>
<tr>
<td>Address</td>
<td>Pt Address</td>
</tr>
<tr>
<td>City</td>
<td>Pt City</td>
</tr>
<tr>
<td>State</td>
<td>Pt State</td>
</tr>
<tr>
<td>Zip</td>
<td>Pt Zip</td>
</tr>
<tr>
<td>Primary Phone #</td>
<td>Pt Primary Phone #</td>
</tr>
<tr>
<td>Age</td>
<td>Pt Age</td>
</tr>
<tr>
<td>Medical Record #</td>
<td>Pt Medical Record #</td>
</tr>
<tr>
<td>Date of Last Visit</td>
<td>Pt Date of Last Visit</td>
</tr>
<tr>
<td>DOB</td>
<td>Pt Birth Date</td>
</tr>
<tr>
<td>SSN</td>
<td>Pt SSN</td>
</tr>
</tbody>
</table>

Rule 6 - No Calculated fields
a. Since we have the Patient’s Birthday we can have Access calculate their actual age at the time of the visit
b. We can also derive and calculate the Date of Last visit by looking for the Maximum date that a patient attended in the Appointment table.

<table>
<thead>
<tr>
<th>Pt First Name</th>
<th>Pt First Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pt Last Name</td>
<td>Pt Last Name</td>
</tr>
<tr>
<td>Pt Address</td>
<td>Pt Address</td>
</tr>
<tr>
<td>Pt City</td>
<td>Pt City</td>
</tr>
<tr>
<td>Pt State</td>
<td>Pt State</td>
</tr>
<tr>
<td>Pt Zip</td>
<td>Pt Zip</td>
</tr>
<tr>
<td>Pt Primary Phone #</td>
<td>Pt Primary Phone #</td>
</tr>
<tr>
<td>Pt Age</td>
<td>Pt Age</td>
</tr>
<tr>
<td>Pt Medical Record #</td>
<td>Pt Medical Record #</td>
</tr>
<tr>
<td>Pt Date of Last Visit</td>
<td>Pt Date of Last Visit</td>
</tr>
<tr>
<td>Pt Birth Date</td>
<td>Pt Birth Date</td>
</tr>
<tr>
<td>Pt SSN</td>
<td>Pt SSN</td>
</tr>
</tbody>
</table>

Rule 7 - Primary Key
a. There are two candidate keys showing here. Pt Medical Record # and Pt SSN. Either should make a good key because they should have no duplicates and no blanks. However, since the medical record number is one we are providing, it’s the safer of the two.
Since we have to do the data entry as well as the database development today, we are going to eliminate some of the extra fields. (keep in mind that these would normally be left in)

<table>
<thead>
<tr>
<th>Pt Medical Record #</th>
<th>Pt Med Rec #</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pt First Name</td>
<td>Pt First Name</td>
</tr>
<tr>
<td>Pt Last Name</td>
<td>Pt Last Name</td>
</tr>
<tr>
<td>Pt Address</td>
<td>Pt Phone #</td>
</tr>
<tr>
<td>Pt City</td>
<td>Pt Birth Date</td>
</tr>
<tr>
<td>Pt State</td>
<td></td>
</tr>
<tr>
<td>Pt Zip</td>
<td></td>
</tr>
<tr>
<td>Pt Primary Phone #</td>
<td></td>
</tr>
<tr>
<td>Pt Birth Date</td>
<td></td>
</tr>
</tbody>
</table>

One Table down, one to go. Our Appointments table will be a scheduling table. What sorts of things would we need to schedule an appointment?

1. Does everything in our list belong to just the scheduled appointment? There are no calculated fields, but we can derive the patient data
   a. The Patient’s information can be found in the Patient Table. If we put in the Patient’s key field, we can link it back to all of the Patient data.

<table>
<thead>
<tr>
<th>Patient’s Name</th>
<th>Pt Medical Record #</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pt First Name</td>
<td>Doctor</td>
</tr>
<tr>
<td>Pt Last Name</td>
<td>Date/Time</td>
</tr>
<tr>
<td>Pt Address</td>
<td>Location</td>
</tr>
<tr>
<td>Pt City</td>
<td>Reason</td>
</tr>
<tr>
<td>Pt State</td>
<td>Type</td>
</tr>
</tbody>
</table>

2. Is our data broken down into its smallest logical parts?
   a. Date and Time can be put in the same field, but for data entry I find it easier to have separate fields.
   b. Location will be fairly standard, so we can make a list of possibilities. If we wanted, we could even make a table of locations to include other things such as the full proper name, the address, even a map that could be used to generate reminder letters.
   c. Type could be made into a choice of First Visit, Walk-in, Emergency, and Physical. If it's possible for more than one of these to be selected, consider using Y/N check box for each option.
3. Are we using unique field names?
   a. Date and Time are a little vague, is that the date of the appt, the date of the follow-up, the date it was scheduled on…? Adding APPT in front of all the fields should make them clearer. We won’t add it to the Patient Med Rec number though, because this value is coming from the Patient Table.

4. Set a primary key.
   a. It’s a common beginner mistake to try to make the Pt Medical Record # the key. If we did this then we would only be allowed to have one visit per person. If someone wanted to come to another visit we would need to create a whole new medical record for them.
   b. There is no real uniqueness here. We could try to do a combination of Patient, Date and Time as a key. It’s possible to do this in Access, but it’s better to have just one field be the key. When there isn’t an inherent key, we use an AutoNumber to create uniqueness for us.
This is the Database Window. The Tables, Queries, Forms, and Reports are kept in the left navigation panel.

You can change the panel to show only one set at a time, or even related sets.

The panel can be collapsed to provide more room on the screen.

Everything we will play with in Access today will have two views. A Data view and a Design view.

The design view is where we build the structure of how the data will show. The data view displays the data saved in the database.

We are only going to play with the design view of the Tables and Queries, but know you can design forms and reports. See the Intro to Forms and Reports packet for more information on those.

When you are in the data view of an object you should have these navigation buttons at the bottom of the window (see previous slide).

Some of the keyboard navigations options are listed here as well.

Tables are the true heart of your database. This is where all of your data lives. Without tables you don’t have a database. All Queries, Forms and Reports pull from the data saved in your tables.

Again we have two views, the Data View and the Design View. The Design view is where we build the structure of our table. The names of the fields/columns, the type of data than can be saved, the lookup values, the number of characters we can type.

The Data View is where we can do data entry, sorting, filtering, editing, and deleting.
### Introducing Queries
- A means of asking questions (querying) of your data
- Can look across a number of tables and other Queries
- Can perform Calculations and Combine fields

Queries are here to help us pull out the sets of data we want from one table or several related tables. This is also where we can do our calculations and derivations.

In the design view we choose which tables and which fields we want to view, in the Data View we see just that set of data.

Depending on the query you may be able to add, edit and delete data from the data view of the query.

### Introducing Forms
- A friendlier view of the database
- Used for data input, menus, display and printing
- Can perform Calculations and Combine fields

Forms are the “pretty” way to put data into the database. Their main purpose is for data entry, and when possible should be where all the data entry is done.

Forms can also be used to create menus. These can be used to create the “front end” of our database, the “user friendly” side.

### Introducing Reports
- Output of information in a printed report
- Allows you to group and summarize data
- Can perform Calculations and Combine fields
- Cannot Edit Data
- Can Make Labels

Reports are nice printouts of our data sets. We can print tables and queries, but they come out as large grids. We can print forms but that usually brings a lot of extra boxes and colors with it. Reports are here to help turn the data into INFORMATION.

Reports can group out data, you can see here it’s doing it alphabetically by patient.

You can do calculations and combinations here as well. We can even make labels.
So here’s how they all tie in together.

Tables are the core of our database. Everything flows to and from the tables. They are our true base of data. Tables are where the data “lives”.

Queries allow us to pull out chunks of data. We can limit the information we see by pulling out specific fields, in the order we want and setting criteria to further limit the results. Queries can pull from one or more tables. Most queries are updatable, meaning you can add and edit the data that is displayed. When you change the data it is SAVED in the tables. When you save a query you are saving the structure for how the data will be pulled from the table. All data lives in the tables. Because of this, our queries will always show the newest data.

Forms are our pretty data entry view. A form can pull its data from a Table or a Query. If you pull from a table you get everything in that table, if you need to limit that information (i.e. only the patients who have been in the clinic in the past two years…) then you pull from a query. Either way, the form is just a structure to display the data in a user friendly manner. All changes to the data, whether pulled from a table or a query, are saved back to the table.

Reports are our neat printable outputs. A report can pull data from a table or a query. Reports are not updateable. If you see a mistake on a report, you need to go back to the record in the table, query or form to make the correction. The next time you run (open) the report it will display the correction.

**Create the Database**

1. Open Microsoft Access

2. Click on Blank Database button on the top center panel

3. In the bottom right corner, click on little yellow folder to choose the Desktop

4. Use the FILE NAME: Patient Appointments
5. Click CREATE to make our Database
   a. This automatically creates a new table

6. Close the new table that is created

**Create the Patients Table**

1. Click on the Create Tab and choose TABLE DESIGN

2. Type the first Field Name: **Pt Med Rec #**

3. Look at List of DATA TYPEs
   a. Text – Alphanumeric characters up to 255 (including spaces)
   b. Memo – Alphanumeric characters up to 64,000 (that’s more than 10 pages at 10 pt font)
   c. Number – True number fields, no letters. A number is data we would do math with.
   d. Date/Time – For Dates and Times
   e. Currency – Special Number field with $
   f. AutoNumber – Automatic numbers created with each new record (we’ll use with the Appt table)
   g. Yes/No – True Boolean - Yes or No - No maybes, no blanks. True or False, Yes or No, On or Off
   h. OLE Object – Object Link or Embed. (holdover from 2003, use attachments or hyperlinks instead)
   i. Hyperlink – Link to a webpage or to a file on your computer
   j. Attachments – Allows you to attach an outside file, much more efficiently than OLE (new to 2007)
   k. Calculated – Allows you to create a calculation based on the data (new to Office 2010)
   l. Lookup Wizard… – We’ll use this feature later to create a combo box like this Data Type List

4. Choose Text for the Pt Med Rec #. Even though we are only going to use numerical values, we will never use this number in a mathematical equation. And by making it a text field we will be able to use hyphens to separate the numbers.

5. Write in a Description of “Patient’s Medical Record Number”, spell out the whole thing

6. Enter in the rest of the fields (descriptions not necessary):

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pt Med Rec #</td>
<td>Text</td>
</tr>
<tr>
<td>Pt First Name</td>
<td>Text</td>
</tr>
<tr>
<td>Pt Last Name</td>
<td>Text</td>
</tr>
<tr>
<td>Pt Phone #</td>
<td>Text</td>
</tr>
<tr>
<td>Pt Birth Date</td>
<td>Date/Time</td>
</tr>
</tbody>
</table>

7. Set the Pt Med Rec # to be the key
   a. Click on the first line (Pt Med Rec #)
   b. Click on the big yellow key on the toolbar

8. Click on the Save button on the quick access toolbar (little floppy disk); Save as Patients

9. Notice the grid button to the left of the Primary Key on the Ribbon. Click this button for the Data View.

10. In the Data view, the grid button is replaced with a little triangle/ruler/pencil button. Click on this button to go back to the data view.
**Entering and Modifying Data**

1. Click on the grid to go to the data, the triangle to go to the design, the grid to go to the data.

2. In the Data view we are blinking on the first and only row. In Excel we are used to seeing all the available rows as soon as we open our file. Here in Access we see only the existing data and one new line. As we add data to our table, Access will automatically go to the next line.

3. Enter our first Med Rec #: 123-456
   a. As soon as you start to type two things happen. A new line appears under the row you are entering and a little pencil appears at the front of the current row, showing we are editing.

4. Press tab move to the next field
   a. First Name: **Shaun**
   b. Last Name: **Fuller**
   c. Birth Date: **1/1/81**
      – Access will add in the "19" for 1981
   d. Phone #: **3525551234**
      – No parentheses, no hyphens

5. Press tab to go to the next record
   a. Pencil will disappear. Shaun Fuller is saved. We could exit Access without doing anything else and Shaun would be here when we got back.

6. Click on the triangle to return to the Design View

7. Move Birth Date above the Phone Number

8. Switch to the Data View

9. Access will ask you to save, click yes.
   a. Data saves itself, but if you change the structure you have to save for the new settings to take effect

10. Enter the next record

<table>
<thead>
<tr>
<th>Pt Med Rec #</th>
<th>Pt First</th>
<th>Pt Last</th>
<th>Pt Birth Date</th>
<th>Pt Phone</th>
</tr>
</thead>
<tbody>
<tr>
<td>789-123</td>
<td>Jacob</td>
<td>Smith</td>
<td>2/2/1972</td>
<td>3525554321</td>
</tr>
</tbody>
</table>

11. Click on the triangle to return to the Design View

12. Click on Pt Birth Date

13. Insert a Row

14. In the new row type Pt Gender, leave as text

15. Switch to the Data View

16. Click in Shaun's Gender box, Type Male (the whole word)

17. Press the down arrow to go to Jacob’s gender and type Male
18. Enter a new record

<table>
<thead>
<tr>
<th>Pt Med Rec #</th>
<th>Pt First</th>
<th>Pt Last</th>
<th>Pt Gender</th>
<th>Pt Birth Date</th>
<th>Pt Phone</th>
</tr>
</thead>
<tbody>
<tr>
<td>555-555</td>
<td>Jennifer</td>
<td>Walton</td>
<td>F</td>
<td>March 3, 1963</td>
<td>352-555-5555</td>
</tr>
</tbody>
</table>

a. Gender is just one character  
b. Date will change back to 3/3/1963  
c. Type in the hyphens for the phone number.

19. Change to the Design view and return to the Data View  
   – Jennifer has moved to the second position, by default Access will sort by the Primary Key Field

20. Let’s set the Gender so it can only accept one character.  
   a. Go to the Design View  
   b. Click on the Gender Field  
   c. At the bottom, in the field properties, change the field size from 255 to 1  
   d. Return to Datasheet view  
   e. Save  
   f. You will get an error message saying data may be lost click Yes to continue

21. Data is lost, our Male entries should now only read M

22. Let’s change the format of our birth dates  
   a. Go to the Design view  
   b. Click on the Birth Date field  
   c. At the bottom, in field properties, change the format to Medium Date  
   d. Return to Datasheet view  
   e. Save

23. Let’s fix the phone numbers  
   a. Go to the Design view  
   b. Click in the Phone Number field  
   c. At the bottom, in Field properties, click in the Input Mask line  
   d. Click the Build button (the ellipsis at the end of the line)  
   e. In the input mask wizard Phone Number is already selected. Click FINISH.  
   f. Access will enter the proper code  
   g. Return to Datasheet view  
   h. Save
24. Shaun and Jacob’s phone numbers look ok, but Jennifer’s looks funny. The input mask didn’t know what to do with her number. If you had a number of these, we could do a find and Replace to remove all the extra characters. Since it’s only the one, click inside Jennifer’s phone number and remove the hyphens. When you move out of that field, her number should be in the correct format.

25. Enter a new record

<table>
<thead>
<tr>
<th>Pt Med Rec #</th>
<th>Pt First</th>
<th>Pt Last</th>
<th>Pt Gender</th>
<th>Pt Birth Date</th>
<th>Pt Phone</th>
</tr>
</thead>
<tbody>
<tr>
<td>369-384</td>
<td>Doris</td>
<td>Jones</td>
<td>F</td>
<td>4/4/54</td>
<td>3525555432</td>
</tr>
</tbody>
</table>

26. Close the Table

**Create Female Patient's Query**

1. Go to the Create Tab

2. Click the Query Design button

3. Click on the ADD button, to add Patients to the query, close the Show Table box

4. Here in the query the top half shows our “data sources” the bottom half, what we are pulling into the query.

5. In the top half we can see our Patient’s table. Pt Med Rec # is bold because it’s our Primary Key

6. We want to see the Patient’s Name and Gender
   a. In the top window, in the Patient list, Double-click on Pt First Name
   b. Double-click on Pt Last Name
   c. Double-click on Pt Gender

7. Click on the Datasheet button on the toolbar
   a. Our data displays in order of Med Rec # even though we can’t see that field

8. Click on the Design button on the toolbar

9. Sort by last name
   a. Under the last name field in the bottom half of the window, change the SORT to be Ascending.

10. Go to the Data View to make sure it looks they way you want (Fuller through Walton)

11. Limit the data to show only the Female Patients
    a. Return to Design view
    b. Under the Gender field, on the CRITERIA line type F
    c. Return to Datasheet view, you should only have two people: Jennifer and Doris

12. Close and save the query as Female Patients
Create Patients Form

1. Go to the Create Tab
   a. Make sure the Table Patients is selected so it becomes the default data source

2. Click on the FORM button

3. We are in the Form's Layout view
   a. Adjust the field widths

4. Change to the "Form" view (first button on the Home Tab)

5. Moving through the Form
   a. Tab, shift-Tab through the fields
   b. Click the navigation buttons at the bottom of the window
   c. Page up and Page down to move between the records

6. Create a new record
   a. Scroll all the way to the end or use the new record button.
   b. Create a new FEMALE patient, anyone you want
      – Careful not to use a Med Rec # that has been used

Note 1: When you tab out of the phone number, everything will disappear. This is a little unnerving at first. But what’s actually happened? We’ve moved to the next record. In the table when we finish a record we move down to the next line, since we can only see one record at a time here, we don’t see the one we just finished. Scroll up and you will see the new record.

Note 2: The structure of the form was created based on the current structure of the table. The birthday is in the same format, the phone number has the input mask, and the gender is limited just one character. While data is always new throughout the database, the “structure” won’t be.

If we decide to go back to our table and insert a middle name column it will not automatically appear here, we would have to go to the design view and add it to the form. If we decided to change the format of the date in the table, it would not automatically change the format of the date here; we would have to do it ourselves.

This is why the planning is so important. The more we plan the less changes we will have to make to the rest of the database. Realize if I add a middle name to the table, I then have to change every query, form and report that I want that middle name to appear in. It’s not impossible, just a pain. However, no matter where you add the new records, be it in the table, query or form it always is saved in the tables, and will appear everywhere it is supposed to.

7. Close and save the form.

8. Open the Patients TABLE, view new person. Open the Female Patients QUERY, view new person. Close Query
Create Reports

1. Go to the Create tab, make sure the Patients table is selected

2. Click the REPORT button
   a. This puts you into layout view; Adjust the columns to fit the data
   b. Right-click to go to the Print Preview
   c. Data sorted in the order it was created in the table
   d. Close and Save as Patients

3. Go to the Create Tab

4. Click the Report WIZARD button
   a. Step 1
      - Choose the Table: Patients
        ⇒ Use double arrow (>>) to move over all fields
   b. NEXT - Step 2 (Grouping) -
      - Group by Name - Double Click on Last Name
        ⇒ Click on Grouping Options button in the bottom left
        ⇒ Change Grouping Intervals to “1st Letter”
        ⇒ Click OK
        ⇒ This will give us the same report we saw in the presentation. Patients grouped by the first letter of their last name
        ⇒ Double click on the blue text on the left to remove it so there is no grouping
      - Group by Date - Double Click on Pt Birth Date
        ⇒ Click on Grouping options to see variations, click Cancel
        ⇒ Double click on Birth Date again, Click on Grouping Options
        ⇒ The same field can be used twice with different grouping options. This is useful for schedules. We can break it down by month, then week, then day, then hour, then minute… As far as we need to go.
        ⇒ Double click on the blue text on the left to remove it so there is no grouping
      - Group by Gender - Double Click on Pt Gender
        ⇒ Make sure that Pt Gender is the ONLY blue text on the left side.
   c. NEXT Step 3 (sorting)
      - We are not going to sort
   d. NEXT Step 4 (layout)
      - Click on each option
        ⇒ Imagine if we had broken it down by the dates. Month, then week, then day…
      - Choose Outline 1
   e. NEXT Step 5 (style)
      - Click on each option
      - Choose your favorite
   f. Step 6 (saving)
      - We shouldn’t save this as Patients since we have modified the structure
      - Change the title of the report to be Patients by Gender
      - Click FINISH
5. You can see there is a little bit of clean up that could be done here, but in general the report is giving us exactly what we asked of it.

6. Right click and go to the layout view, adjust the birthday field

7. Close and save the report

We have made a table, a query, a form and two reports. This is fine; some Access databases only have one table. But the true power of Access is being able to work with multiple tables.

Create Appointments Table

1. Go to the Create Tab

2. Click on Table Design

3. Set Appt ID # to be the Primary Key

4. Look up list in table
   a. Go to the design view of the table,
   b. Change the DATA TYPE of Appt Location to be LOOKUP WIZARD
      (We are no longer typing in text, we are now looking up text.)
      − Step 1. I will type in the values I want
      − Step 2. Shands GNV; Shands JAX; Med Plaza; Magnolia Ctr
      − Step 3. Finish
   c. Appt Location will still show Text as the data type
   d. Switch to Datasheet View

5. Save Table as Appointments

6. Switch to Datasheet view

7. Enter a new record
   a. Tab past the AutoNumber
   b. Med Rec #: 123-456
      − As soon as you start typing, Access will fill in the AutoNumber for you
   c. Appt Doctor: Jekyll
   d. Appt Date: 10/17
      − Access will assume you mean this year
   e. Appt Time: 2p
      − Since this is a date/time field, Access knows you mean 2:00 PM
   f. Appt Location: Choose from list
   g. Appt Reason: Mood Swings
      − After all, we’re seeing Dr. Jekyll
   h. Check Appt Type First

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appt ID #</td>
<td>AutoNumber</td>
</tr>
<tr>
<td>Pt Med Rec #</td>
<td>Text</td>
</tr>
<tr>
<td>Appt Doctor</td>
<td>Text</td>
</tr>
<tr>
<td>Appt Date</td>
<td>Date/Time</td>
</tr>
<tr>
<td>Appt Time</td>
<td>Date/Time</td>
</tr>
<tr>
<td>Appt Location</td>
<td>Text</td>
</tr>
<tr>
<td>Appt Reason</td>
<td>Text</td>
</tr>
<tr>
<td>Appt Type First</td>
<td>Yes/No</td>
</tr>
<tr>
<td>Appt Type Referral</td>
<td>Yes/No</td>
</tr>
</tbody>
</table>
8. Look up Patients  
   a. It would be good to have a list of all of the Patients so we don’t have to look in the Patient table for their Medical Record Numbers  

   b. Go to the design view of the table,  

   c. Change the DATA TYPE of Pt Med Rec # to be LOOKUP WIZARD  
      - Step 1. I want the lookup column to look up the values in a table or query  
      - Step 2. (Patients table is already selected)  
      - Step 3. Bring over: Pt Med Rec, Pt Last Name, Pt Birth Date  
      - Step 4. Sort by: Pt Last Name, Pt Birth Date  
      - Step 5. UNCHECK the hide key column  
      - Step 6. “Choose a field that uniquely identifies the row”. In this case it’s the Pt Med Rec #  
      - Step 7. Finish  

   d. The table must be saved before the RELATIONSHIP can be created  
      - Click Yes  
      - By creating this look up we have made a simple relationship between our two tables  

9. Switch to Datasheet view, enter a new record (anything is fine)  

10. This works beautifully. However, it’s hard to tell who we have chosen from the list of Med Rec #s. It would be nice if we could see their names. We can do this with a query.  

11. Close the table  

Create a Schedule Query  
1. Go to the Create Tab and click the Query Design button  

2. Add both tables and close the show table window  
   a. If you missed one, you can drag the table from the navigation panel  
   b. If you add one too many, right click on the extra table and Choose remove table  
   c. If all else fails, close and try again  

3. From the Appointment table Double Click on Pt Med Rec #  
   a. Yes it matters which table it came from. The Med Rec # in Appointments has a drop down list of all possibilities for us  

4. From the Patient Table Double Click on Pt First Name and Pt Last Name  

5. From the Appointment Table Double click on Appt Doctor, Appt Date and Appt Time  

6. Switch to the Datasheet view  

7. Add a new record - Choose the Med Rec #, Tab over to enter Doctor, Date and Time  

8. This is great, as we choose our Med Rec # from the list, the Patient's name appears automatically. However as that list gets longer; this process will become more difficult.  

9. Close and Save Query as Schedule.
Create Patient Appointment Form
1. We can do data entry in Tables and in Queries. But the best place to do data entry is in the **Forms**. The new query we have created shows us who we have chosen from our list, but as that list gets longer it is going to be much more cumbersome to choose from. There is no way to search for a Patient.

2. Go to the Create Tab

3. In the Forms Group, choose MORE FORMS, choose FORM WIZARD
   a. Choose the Table: Patients
      – Use the Double Arrow to bring over everything (>>)
   b. DO NOT CLICK NEXT
      – notice the top of the window says “You can choose from more than one table or query”
   c. Choose Table: Appointments
      – Bring over: Doctor, Date, Time
   d. Click FINISH - We are skipping the rest of the steps

4. Access will create a form with a SubForm for us. The patient data is at the top, the appointments for that patient are listed underneath.

Using Patient Appointments Form
1. Click in the last name field
   a. Click the binoculars to FIND (or press Ctrl-F)
   b. Type in Walton
   c. Schedule an appt for Ms Walton.

2. Create a new Patient
   a. Schedule them for an appointment
   b. We can do it all on one screen now, and the data will flow back to both tables.

3. Close and Save the form

View the Final Results
1. View each object in the database
   a. Your Tables, Queries, Reports

Database Size
1. Check size of the database
   a. Exit Access
   b. Right-click on file
   c. Select Properties

2. Compact and Repair
   a. Open the database
   b. Open the File Menu, Info Section
   c. Choose Compact and Repair
   d. Check File Size again
Backing up Database

1. Method 1
   a. Open database
   b. Open the File Menu, Save & Publish Section
   c. Select Backup Database
   d. Click Save As button

2. Method 2
   a. Exit out of Microsoft Access
   b. Right Click on file, Send to -> Compressed zipped folder
   c. Check File size. This is how I recommend you do backups.

   Congratulations, you now know enough to be dangerous.