**Name: Session:**

**Programming II**

**Lab Exercise 5/21/2021**

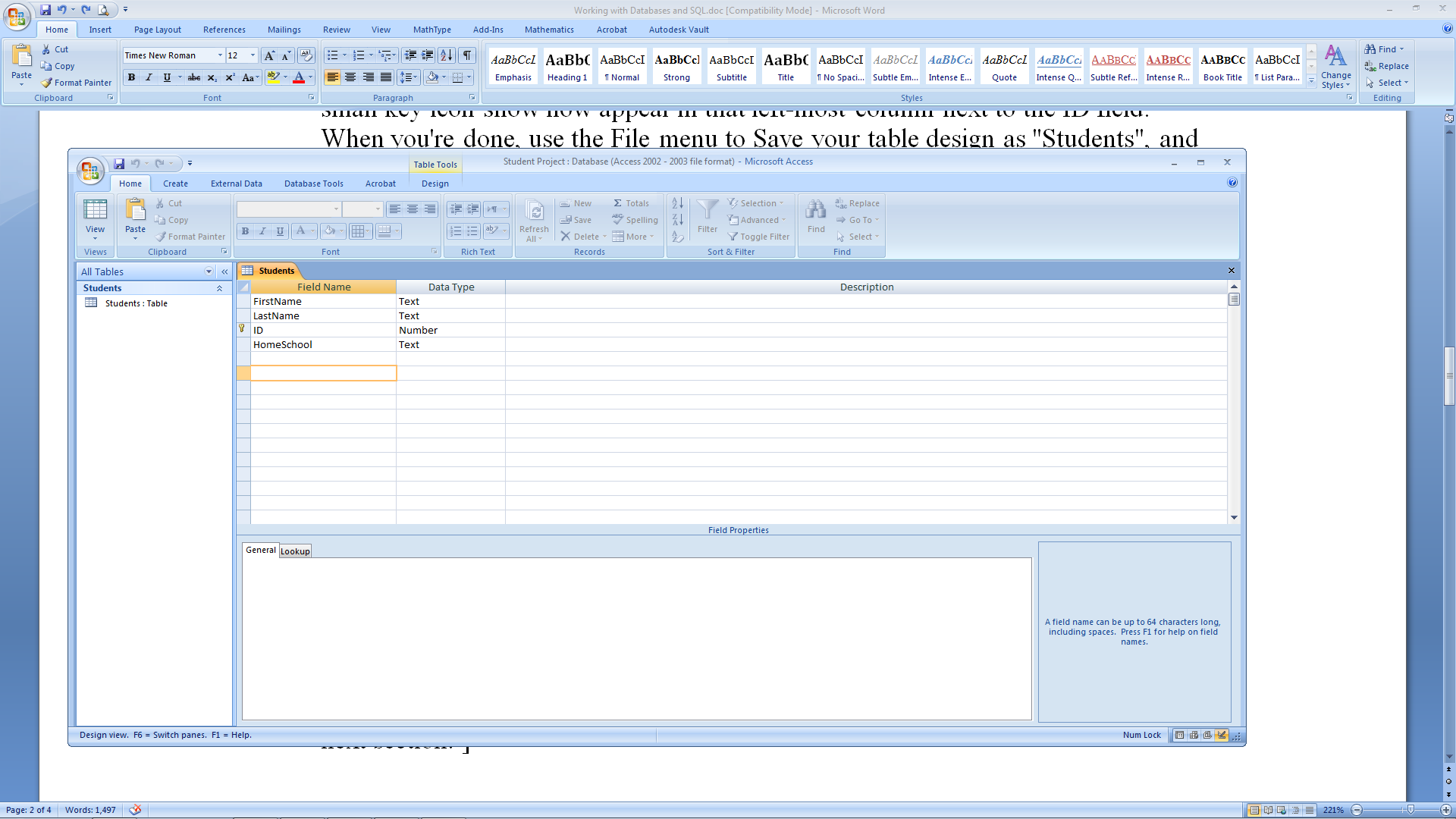
**Working with Databases and SQL**

**Overview**:  now for something completely different.  Before we start database programming, the goal is to gain some experience working with databases and SQL.  We'll do this by creating a database using Microsoft Access, and then use MS Access to execute SQL queries.

**Part 1:  Database Schema**

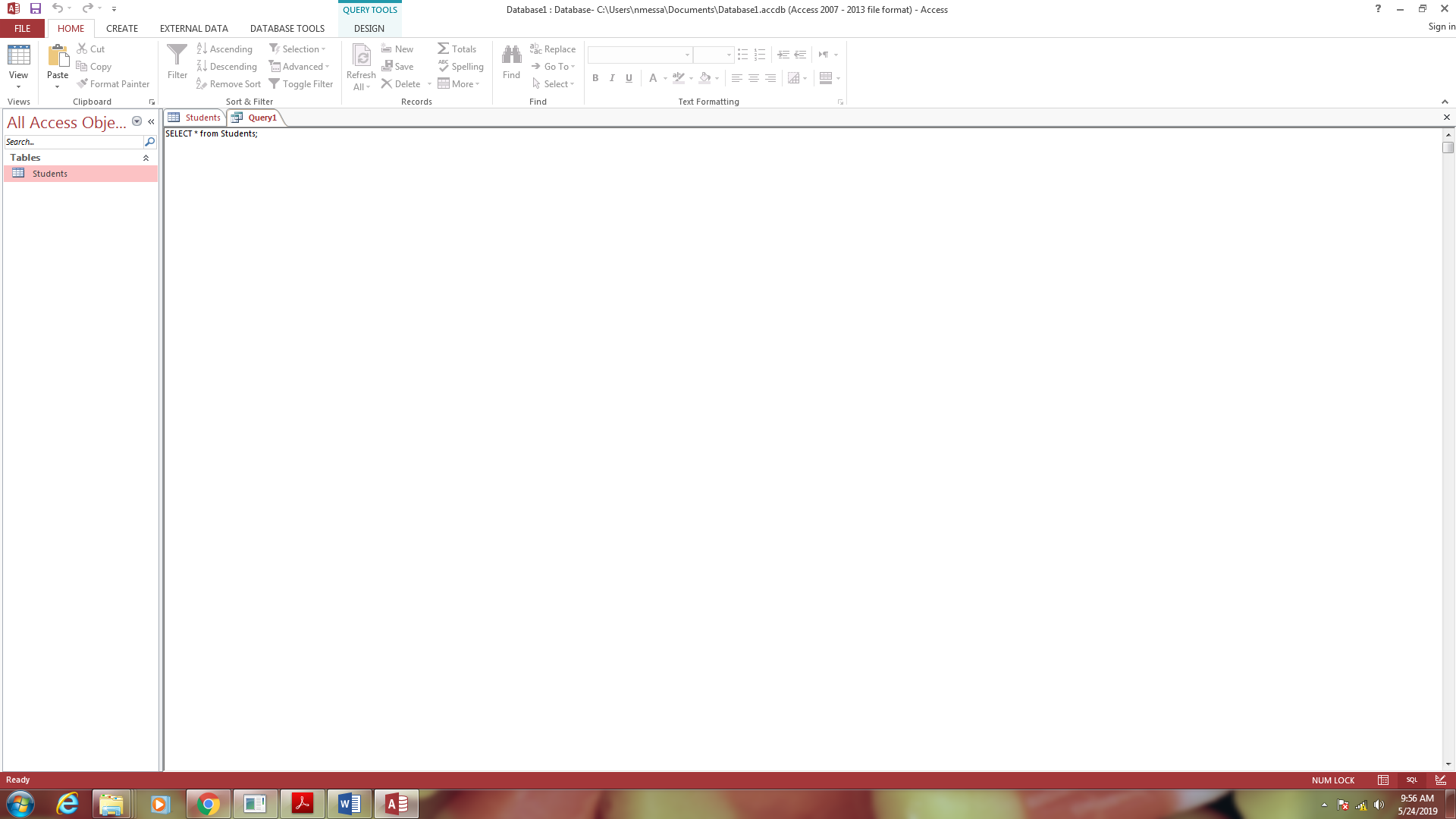
1. You're going to create a database for students in your class.  Our simple database has only one table:  *Students*.  Before we start, let's define the database schema.  The Students table contains the following fields and field attributes:
   * **ID**:  attendee's unique integer id (Number, Long Integer, Primary Key)
   * **FirstName**:  attendee's first name (Text, field size of 64, Required, Indexed (Duplicates OK))
   * **LastName**:  attendee's last name (Text, field size of 64, Required, Indexed (Duplicates OK))
   * **HomeSchool**:  attendee's institution (Text, field size of 64, Required)

**Part 2:  Creating a Database in Microsoft Access**



1. Start Microsoft Access (Start, All Programs, Microsoft Access) and create a New blank database via the File menu.  Use the file name "Student Project.mdb", and then click the Create button.  You should now see something similar to the screenshot above with the exception of Table1. Close Table1.
2. Create a new table in Design View and add the 4 fields needed for the Students table according to the database schema given earlier, in the order shown.  For each field, use the field name given in the schema (*ID*, *FirstName*, etc.).  For Data Type, make sure ID is set to Number, and the rest are Text.  For the ID field, make sure the Field Size attribute (shown at the bottom of the screen) is set to Long Integer, and that Required is set to Yes.  For the fields that are of type Text, set their Field Size to 64, and set Required to Yes.  For the FirstName and LastName fields, also set their Indexed attribute to "Yes (Duplicates OK)".  Finally, click to the left of the ID field to select that entire row, and then make it the table's primary key by selecting Primary Key from the right-click menu; a small key icon show now appear in that left-most column next to the ID field.  When you're done, use the File menu to Save your table design as "Students", and then click the X to close the table designer window.
3. Now you should see the exact same window as shown above:  a database named "Student Project" with a single table named "Students".  Note that you can redesign the table at any time by right-clicking on the table name, and selecting Design View.
4. Okay, now let's add some data to the table.  Double-click on the Students table to add data to it.  You don't need to add much data, say 3-4 records for now. Later, I will show you how to import a text file with the data for all the attendees.  Note that since the ID field is the primary key, each record must have a unique ID value.  The convention is to start at 1, and number each record one larger than the previous:  1, 2, 3, ...  Note that MS Access automatically saves your data as soon as you leave the record, so all you really need to do is tab around the fields of the table, entering data.  After you have added 3-4 records, simply close the table via the X.  To verify that your data is really there, reopen the table by double-clicking on it.
5. At this point, you now have a fully-functional relational database!  [ You will be using Visual Studio.NET instead of MS Access in the next section. ]

**Part 3:  Working with SQL**



1. Okay, let's write a simple SQL Select query for your database.  MS Access is designed to generate queries via drag-and-drop, but we want to program in straight SQL.  MS Access makes this a little harder, but here is how we can do it.
   1. From the Create tab, click Query Design.
   2. Close the Show Table window
   3. Right-click on the Query 1 part of the window and select SQL view.
2. Into this query window type the SQL query "Select \* from Students;".  Now run the query by either clicking the Exclamation mark (!) in the toolbar or by choosing Run from the Query menu.  Presto, you should see your entire table in the Query window!  Close the query window; you don't need to save this simple query, but you can if you want.
3. Now, let's get some real data into your table.
   1. Create text file in Notepad, "students.txt". You file should be in this format:

1,"Lowell","Carmony","Lake Forest College"

2,"Joe","Hummel","Lake Forest College"

3,"Tony","Richardson","University of Evansville"

We are using this comma-separated format because MS Access can easily import data from such a file.

1. To start the import from the External Data tab, select "Text Files."
   1. Now you need to navigate to the folder
   2. Select the text file "students.txt" when you find it, and choose “Append a copy of records to the table” and click the OK button.
   3. The Import Text Wizard window will now appear, with the option button for "delimited" already selected.
   4. Click Next to continue, and you should see the data being separated into columns due to the option button for "Comma" being selected.
   5. Again, click Next to continue.  In what is now the third wizard screen. You should rename your fields to an appropriate name.
   6. In the fourth wizard screen, set the primary key to ID.
   7. In the fifth wizard screen, type in the name of the table where you want your data stored.
   8. Click Finish.  The data from the text file should now be imported; double-click on the Students table in the MS Access window to confirm this.  Also confirm that the right data went into the right fields...
2. Now that we have some additional data to play with, let's write some more interesting SQL queries.  To execute an SQL query, you want to click on Queries along the left-hand side of the MS Access window, double-click on "Create query in Design View", close the dialog window that pops up, and then switch to SQL View via the View menu.  Write SQL to perform each of the following.  Note that you don't have to open a new query window each time, you just need to switch back to SQL View:
3. Repeat your Select query to retrieve all data from the Students table.
4. Write a Select query to retrieve all data from the Students table, but in order by last name.  Note that if two people in the workshop have the same last name, then there is only a 50% chance that the attendees will be properly alphabetized.  You can ensure a more precise result by adding a clause like "Order By LastName Asc, FirstName Asc" to the end of your query.  Repeat but sort by first name, last name.
5. Write a Select query to retrieve all data, but sorted in descending order by last name, first name.  [ Hint:  ASC means ascending in SQL, while DESC means descending. ]
6. Write the query "Select Count(\*) As TOTAL from Students;" to determine the total number of attendees in the workshop.
7. Write a query to obtain the maximum ID in the Students table.  To get you started:  "Select Max(ID) As MaxID ...".
8. Write a query to see if anyone else has the same last name as you.  What institution are they from?  Anyone with the same first name?
9. Write a query to retrieve the name of everyone at the workshop from your institution. Note that the way to do this is to add a " Where Institution = 'your institution name here' " clause to the end of your query (without the ").  Note the use of the single quotation marks, which are required by SQL to delimit text values in a query; SQL will also let you use double quotations marks " to delimit text values in a query, but " will lead to problems later on when we start programming, so it's better to use ' quotes.  In general, the ' delimiter is required around text and data values in SQL.