­­

CARRIER DOME TICKET RESERVATION SYSTEM

IST659 – M003

EASHANI DEORUKHKAR

JEET GANATRA

**Project Summary**

Syracuse University is renowned for its athletic events. The men's and women's basketball teams, the football team, and both the men's and women's lacrosse teams play in the Carrier Dome which is the largest domed stadium of any college campus in the US. It has a capacity of 49250.

This project is being designed to streamline ticketing services for Syracuse University’s Carrier Dome. Each game that takes place at the dome usually has a huge crowd turnout and keeping track of all the tickets that are sold at these games becomes a huge task. The tickets are currently only sold at ticket booths which becomes inconvenient as the customers have to physically go and buy tickets.

Our system combines the offline system with a way to buy tickets online too. There will be an interface online where tickets can be bought and the data regarding it will be saved in the same database itself.

There are tickets sold the week before each event which can be bought at ticketing booths which are open from 8:00 AM to 5:00 PM weekdays or online any time up to the day before the event. A customer (who can either belong to SU or not) buys a ticket. While a customer needs to buy at least one ticket, the maximum number of tickets that can be bought are 5. This ticket is sold by one of the employees at the booth or online. The ticket can belong to 3 categories (level 1, level 2, level 3) and the prices differ based on the category. The ticket prices can also differ based on the event happening which could be a football or basketball game, band championships or cheerleading competitions.

The maximum number of tickets that can be sold depends on the capacity of the stadium i.e 49250. This value does not change. The ticket gets assigned to the customer after it is bought. A ticket once bought cannot be refunded. Payments can be made with cash or card but card transactions will cost $1 more.

This system will record information about ticket numbers, the customer, the transactions where the tickets are sold, the event that the tickets are for, the seat type, the employees responsible for the transaction(if bought at booth) and the booths where the tickets are sold.

There is a lot of information that needs to be kept track of and this database needs to be designed such that all the people responsible for managing the various sections of the event have appropriate access to it.

**Technologies used**

**MS Access:**

Microsoft Access is a database management system (DBMS) offered by Microsoft that allows for GUI access of databases. Data can be created in access or imported from other sources

**MS Visio:**

Microsoft Visio is a diagram creation tool that can be used to create flow diagrams and ERDs.

**SQL:**

Structured Query Language is a language used to work with relational databases. It also has an extension called PL-SQL for procedural logic.

ENTITY AND ATTRIBUTE TABLE

1. Customer table- captures information about the customers who buy tickets

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ENTITY NAME: CUSTOMER | ATTRIBUTE NAME | FIELD TYPE | NULL/NOTNULL | EXPLANATION |
| PRIMARY KEY | Cust\_ID | int | NOTNULL | Unique id of customer |
| OTHER ATTRIBUTES | CFname | Varchar(20) | NOTNULL | First name of customer |
|  | Clname | Varchar(20) | NULLABLE | Last name of customer |
|  | SUID | Int(8) | NULLABLE | SUID if applicable |
|  | PHONE | Int(10) | NOTNULL | Phone of customer |

1. employee table- captures information about the employees who work at the Dome

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ENTITY NAME: EMPLOYEE | ATTRIBUTE NAME | FIELD TYPE | NULL/NOTNULL | EXPLANATION |
| PRIMARY KEY | EmpID | int | NOTNULL | Unique id of employee |
| OTHER ATTRIBUTES | EFname | Varchar(20) | NOTNULL | First name of employee |
|  | Elname | Varchar(20) | NULLABLE | Last name of employee |
|  | EDesig | Varchar(8) | NULLABLE | Designation of employee |

1. event table- information about the possible events that take place at the dome

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ENTITY NAME: EVENT | ATTRIBUTE NAME | FIELD TYPE | NULL/NOTNULL | EXPLANATION |
| PRIMARY KEY | EventID | int | NOTNULL | Unique id of event |
| OTHER ATTRIBUTES | name | Varchar(20) | NOTNULL | Name of event |
|  | date | Datetime() | NOTNULL | Date and time it was held |

1. booth table- information about the booths where

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ENTITY NAME: BOOTH | ATTRIBUTE NAME | FIELD TYPE | NULL/  NOTNULL | EXPLANATION |
| PRIMARY KEY | BoothID | int | NOTNULL | Unique id of booth |
| OTHER ATTRIBUTES | Tick\_sold | Int(10) | NOTNULL | Number of tickets sold |
|  | empID | Datetime() | NOTNULL | Date and time it was held |

1. Seat table- information about seat type

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ENTITY NAME: SEAT | ATTRIBUTE NAME | FIELD TYPE | NULL/  NOTNULL | EXPLANATION |
| PRIMARY KEY | seatcode | int | NOTNULL | Unique id of seat type |
| OTHER ATTRIBUTES | seattype | Varchar(10) | NOTNULL | Type of seat |
|  | price | Int | NOTNULL | Price of seat according to type |

1. ticket table – information about tickets

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ENTITY NAME: TICKET | ATTRIBUTE NAME | FIELD TYPE | NULL/NOTNULL | EXPLANATION |
| PRIMARY KEY | Tick\_ID | int | NOTNULL | Unique id of ticket |
| OTHER ATTRIBUTES | Cust\_ID | int | NOTNULL | Unique id of customer |
|  | Event\_ID | int | NOTNULL | Unique id of event |
|  | Seat\_code | Int | NOTNULL | Code of seat type |
|  | Booth\_ID | Int | NOTNULL | Unique id of booth |
|  | Emp\_ID | int | NULLABLE | Id of employee who sold ticket |
|  | price | int | NOTNULL | Price of ticket |

ER Diagram



Creation of tables

CREATE TABLE Customers(

      CustomerID int NOT NULL PRIMARY KEY,

      CFname varchar(30) NOT NULL,

      CLname varchar(30) NOT NULL,

      Cphone varchar(20) NOT NULL,

      CSUID varchar(30),

);

CREATE TABLE Employees(

      EmpID int NOT NULL PRIMARY KEY,

      EFname varchar(30) NOT NULL,

      ELname varchar(30) NOT NULL,

      Edesignation varchar(30) NOT NULL,

);

CREATE TABLE Seat(

      Seatcode varchar(20) NOT NULL PRIMARY KEY,

      Seattype varchar(30) NOT NULL,

      Seatprice varchar(30) NOT NULL);

CREATE TABLE Event(

      EventID int NOT NULL PRIMARY KEY,

      Eventname varchar(30) NOT NULL,

      Eventdate datetime NOT NULL,

CREATE TABLE Booth(

      BoothID int NOT NULL PRIMARY KEY,

      Ticketsold varchar(30) NOT NULL,

      EmpID int NOT NULL

      FOREIGN KEY (EmpID) references Employees(EmpID)

      );

CREATE TABLE Tickets(

      TicketID int NOT NULL,

    CustomerID int NOT NULL,

      EmpID int NOT NULL,

      EventID int NOT NULL,

      Seatcode varchar(20) NOT NULL,

      BoothID int NOT NULL,

      FOREIGN KEY (CustomerID) REFERENCES Customers(customerID),

      FOREIGN KEY (EmpID) REFERENCES Employees(EmpID),

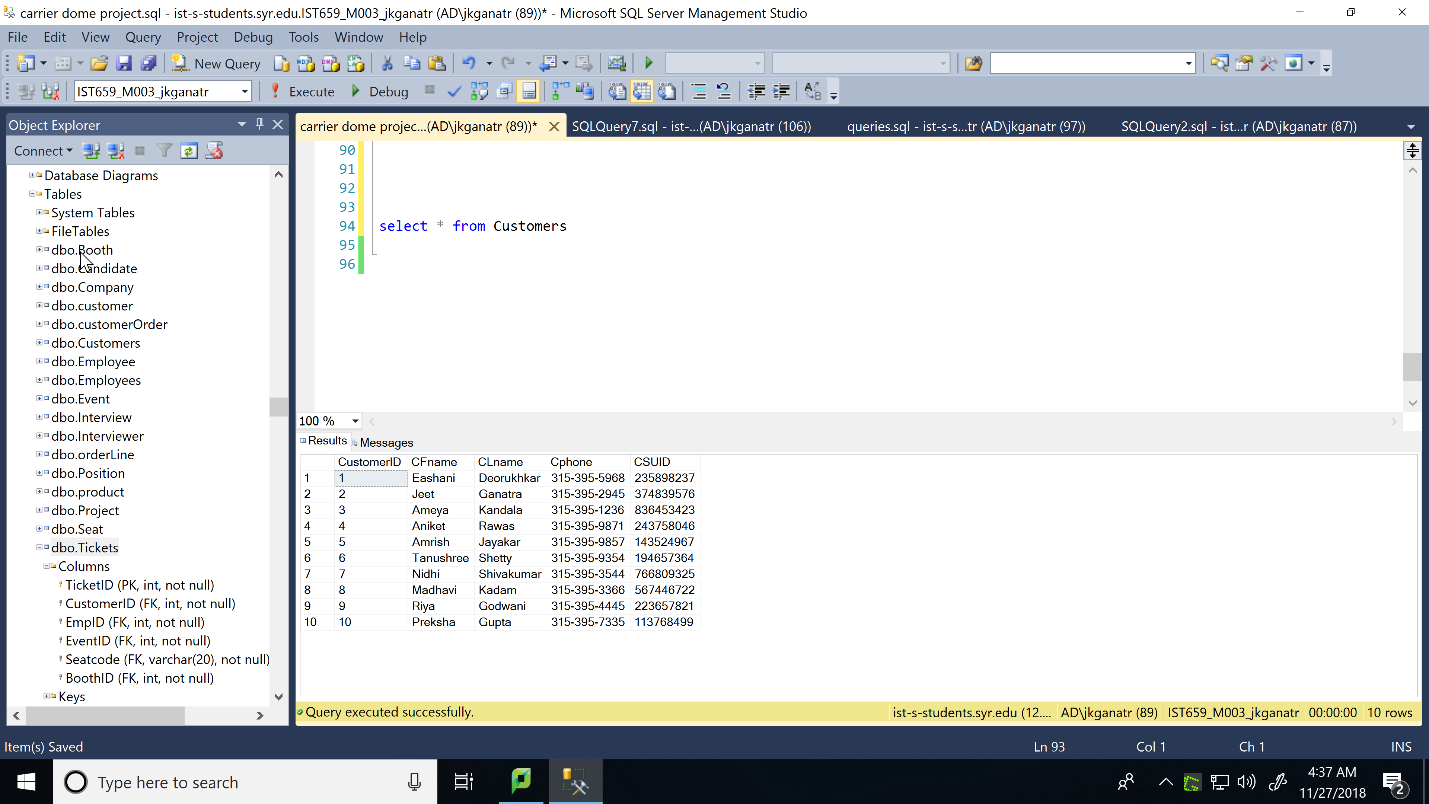
      FOREIGN KEY (EventID) REFERENCES Event(EventID),

      FOREIGN KEY (seatcode) REFERENCES Seat(seatcode),

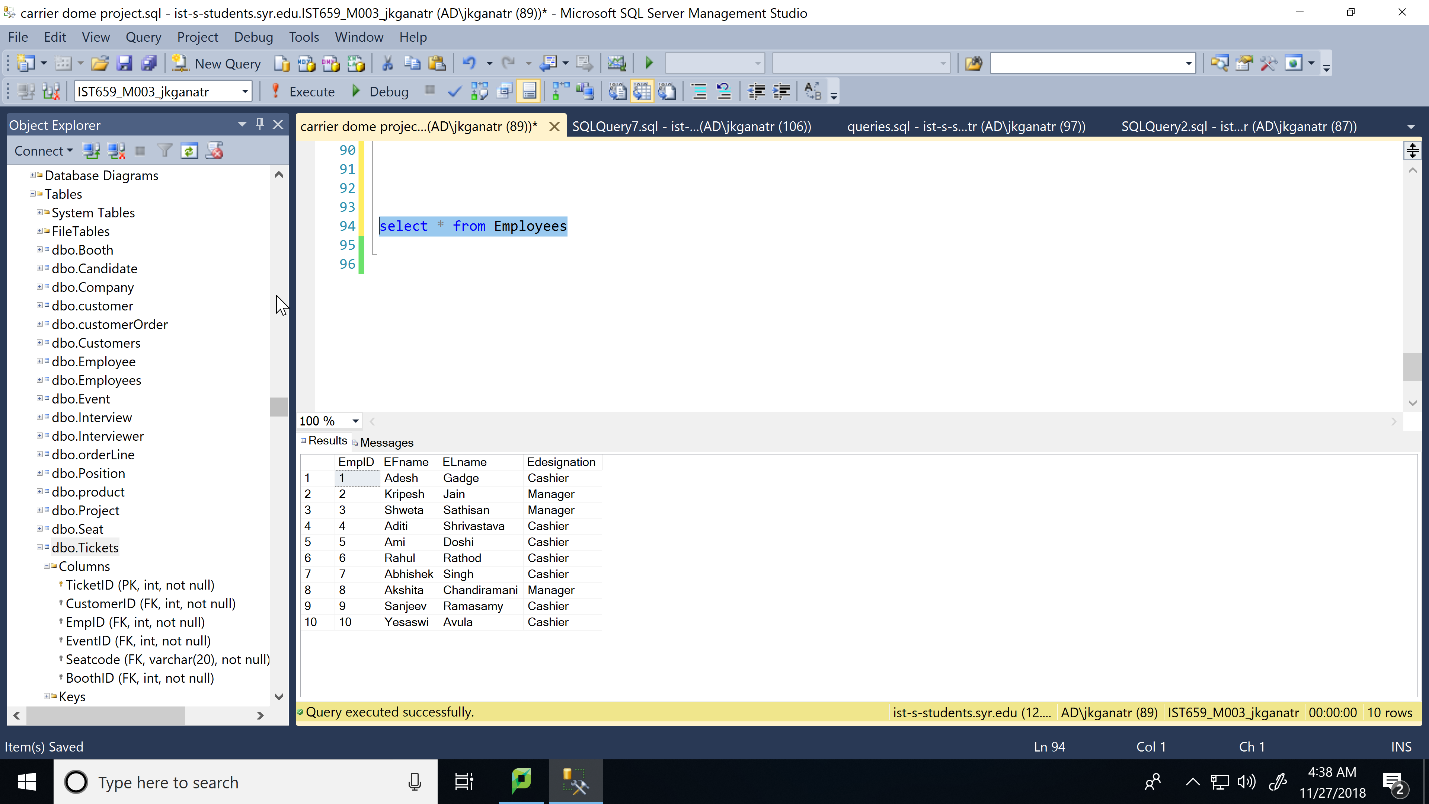
      FOREIGN KEY (BoothID) REFERENCES Booth(BoothID));

 Tables after data insertion

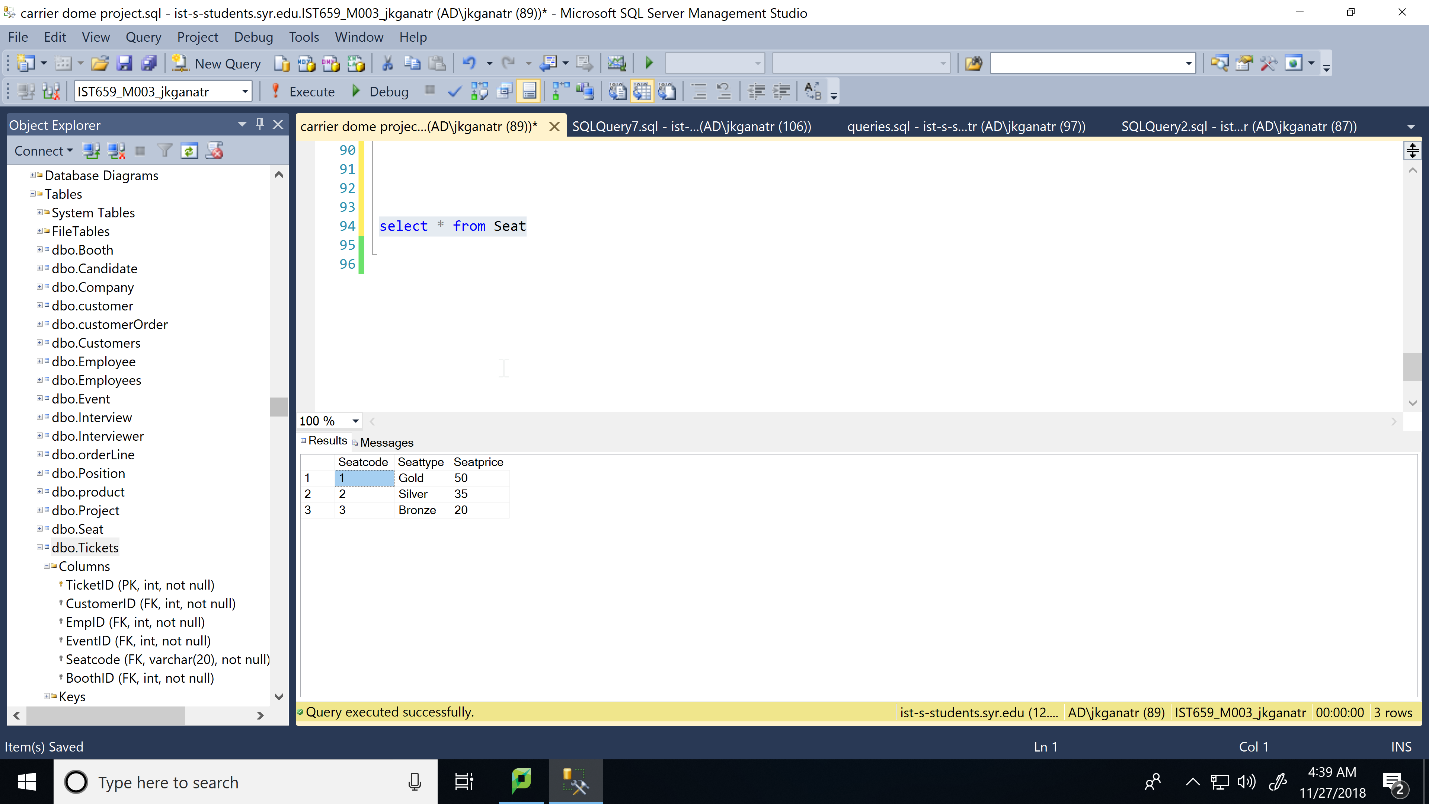
Customers table



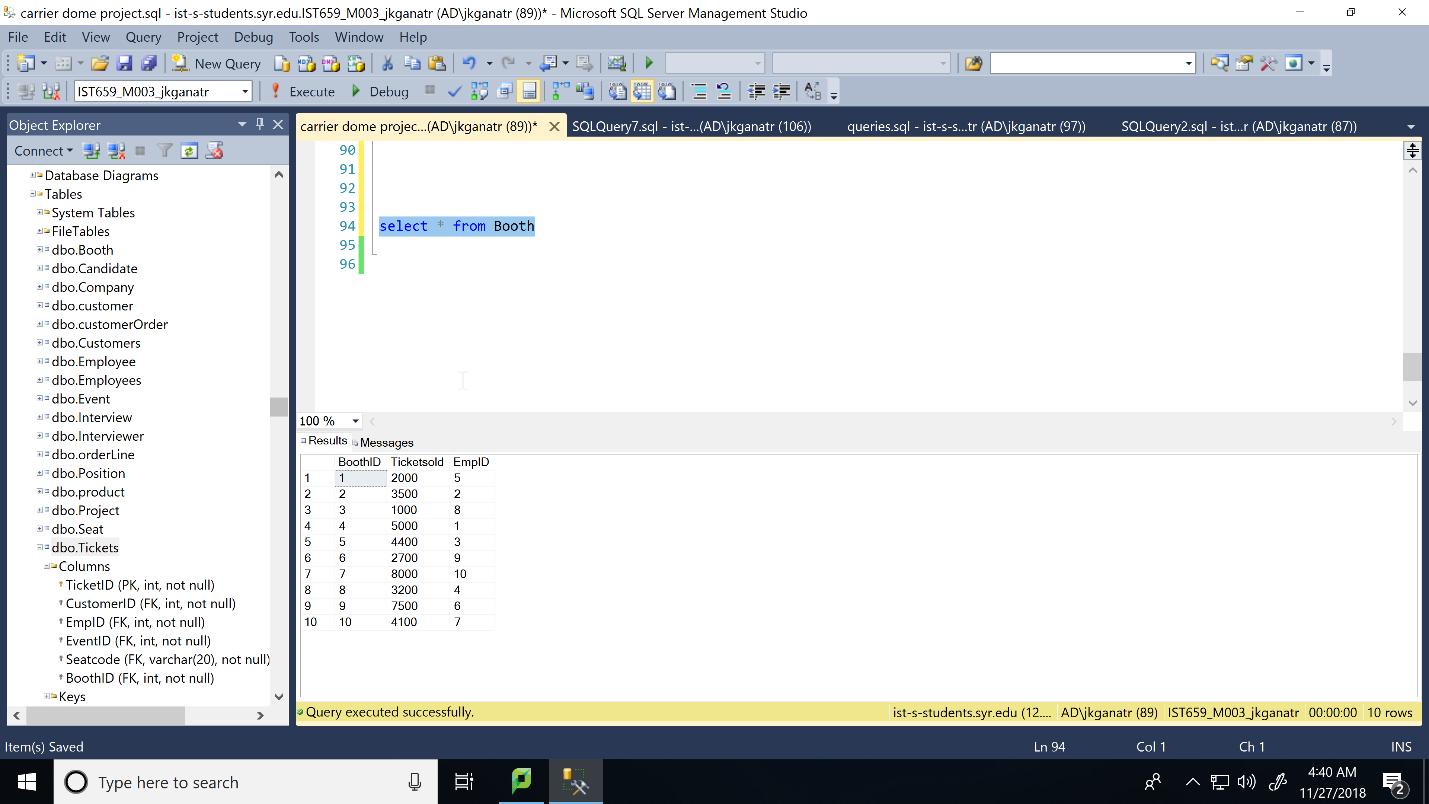
Employees table



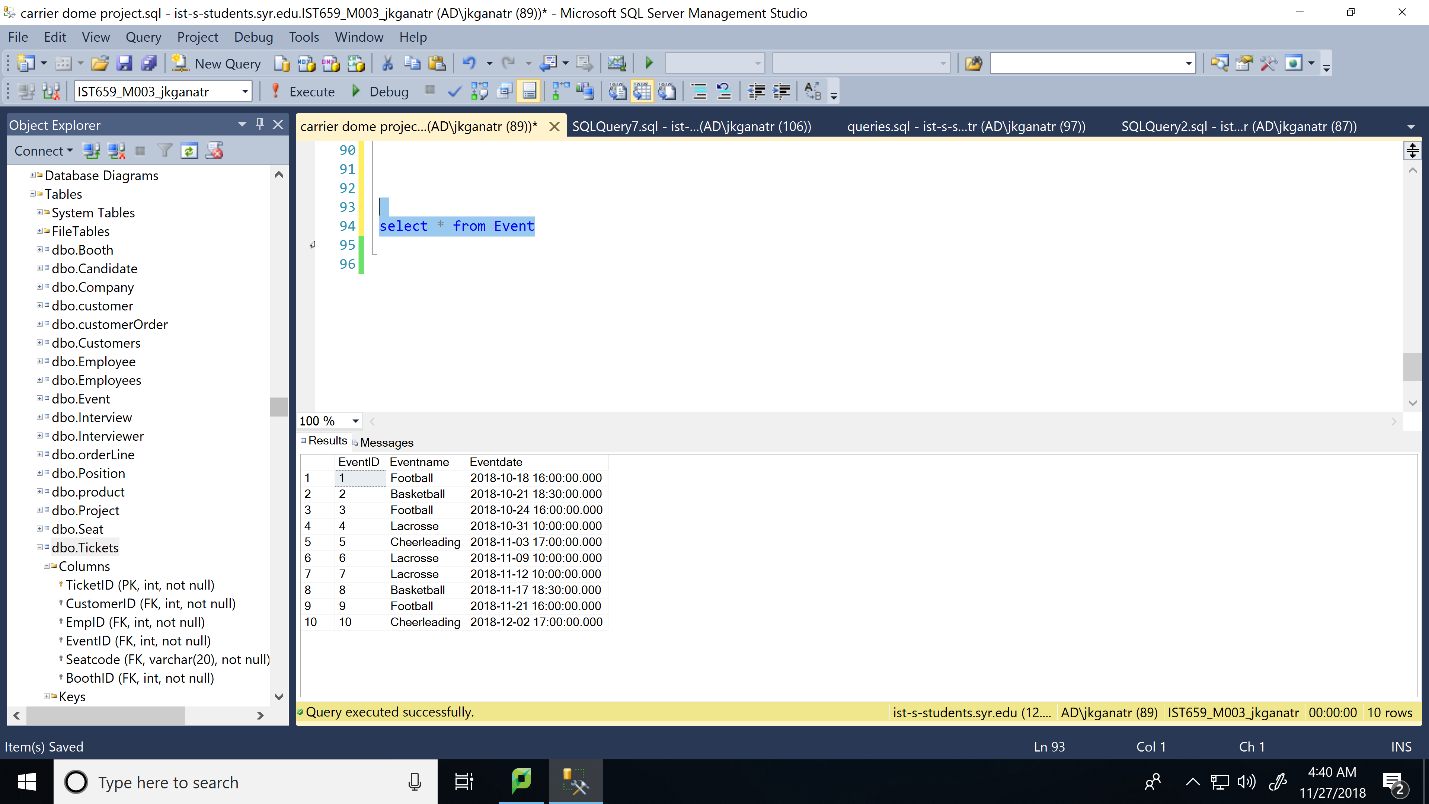
Seat table



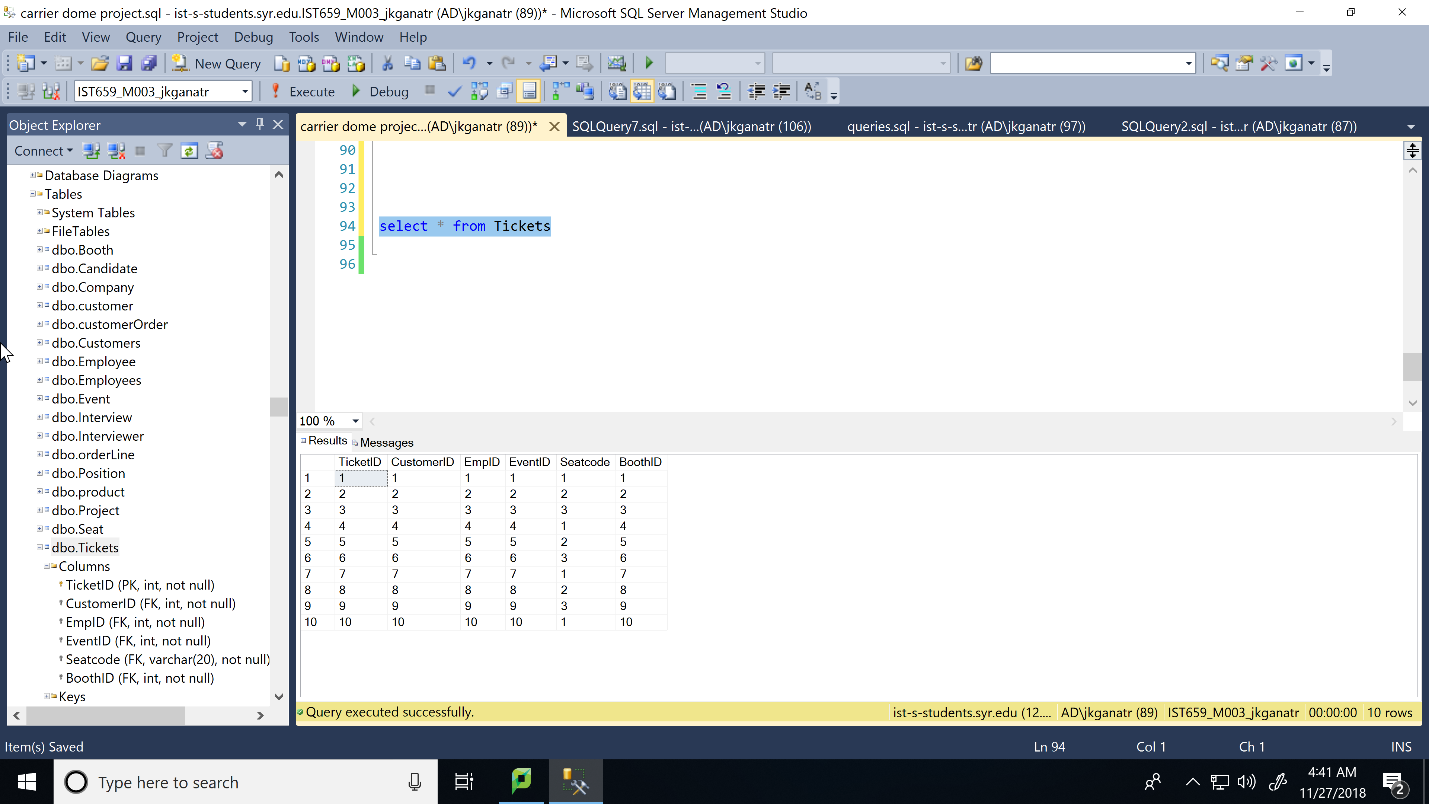
Booth Table



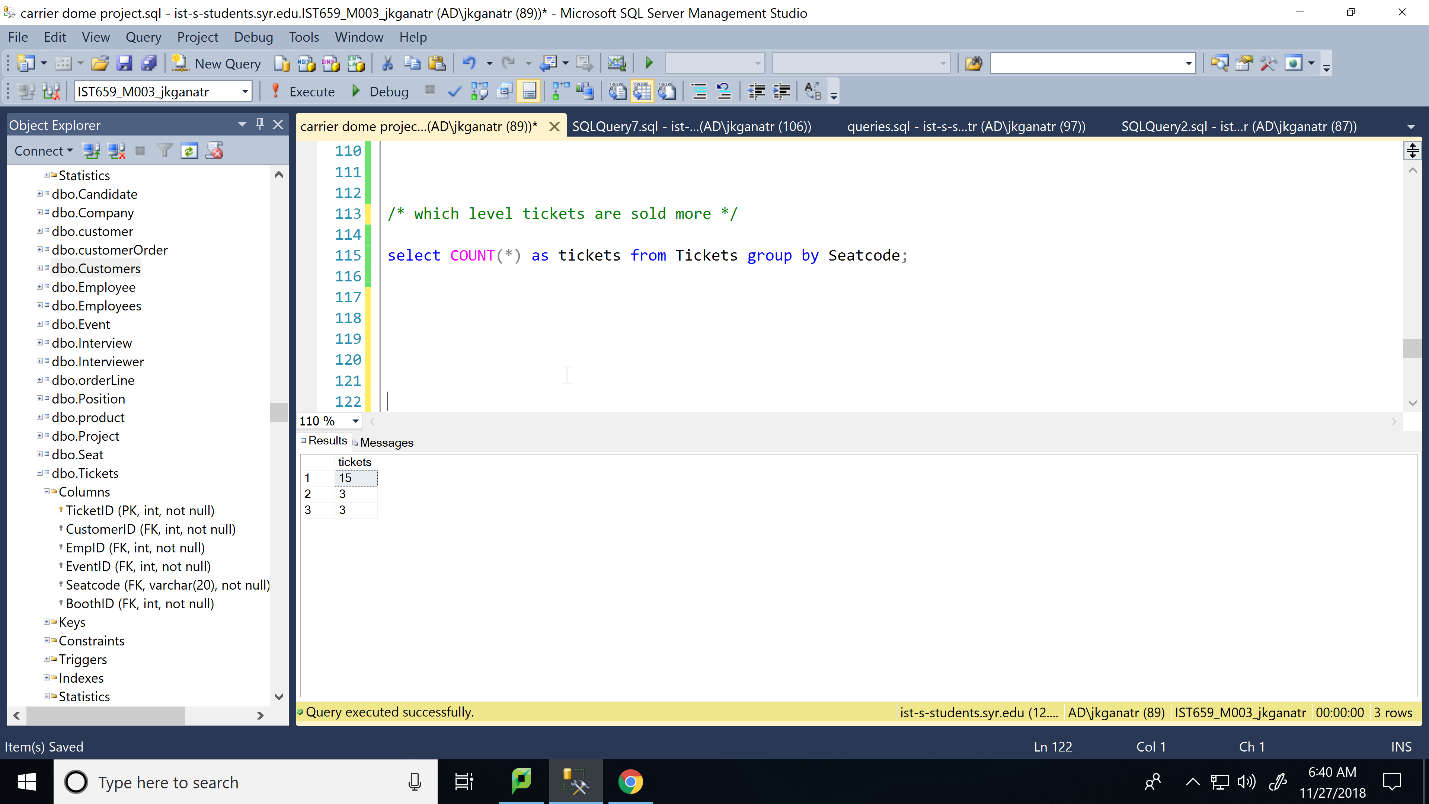
Event table

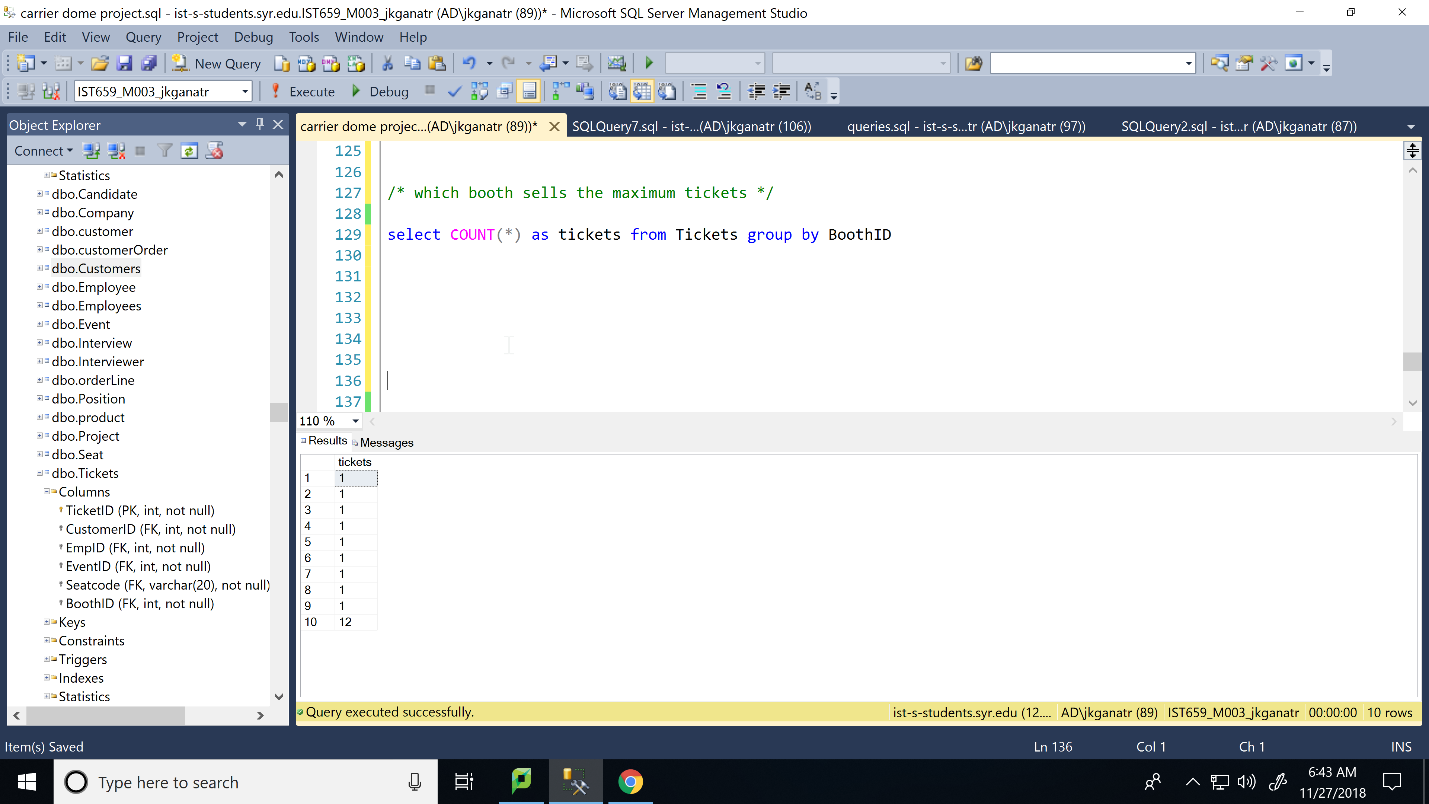


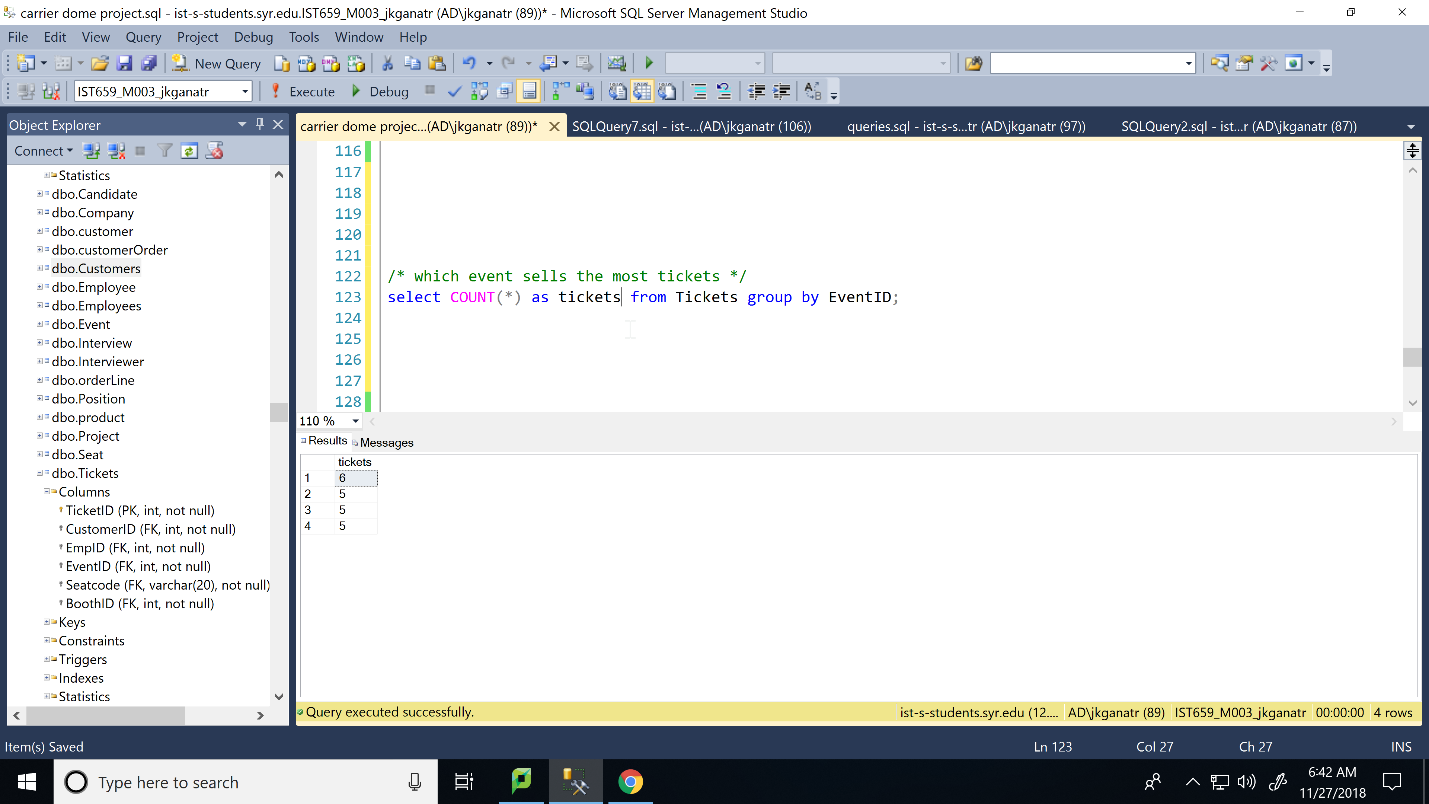
Tickets table

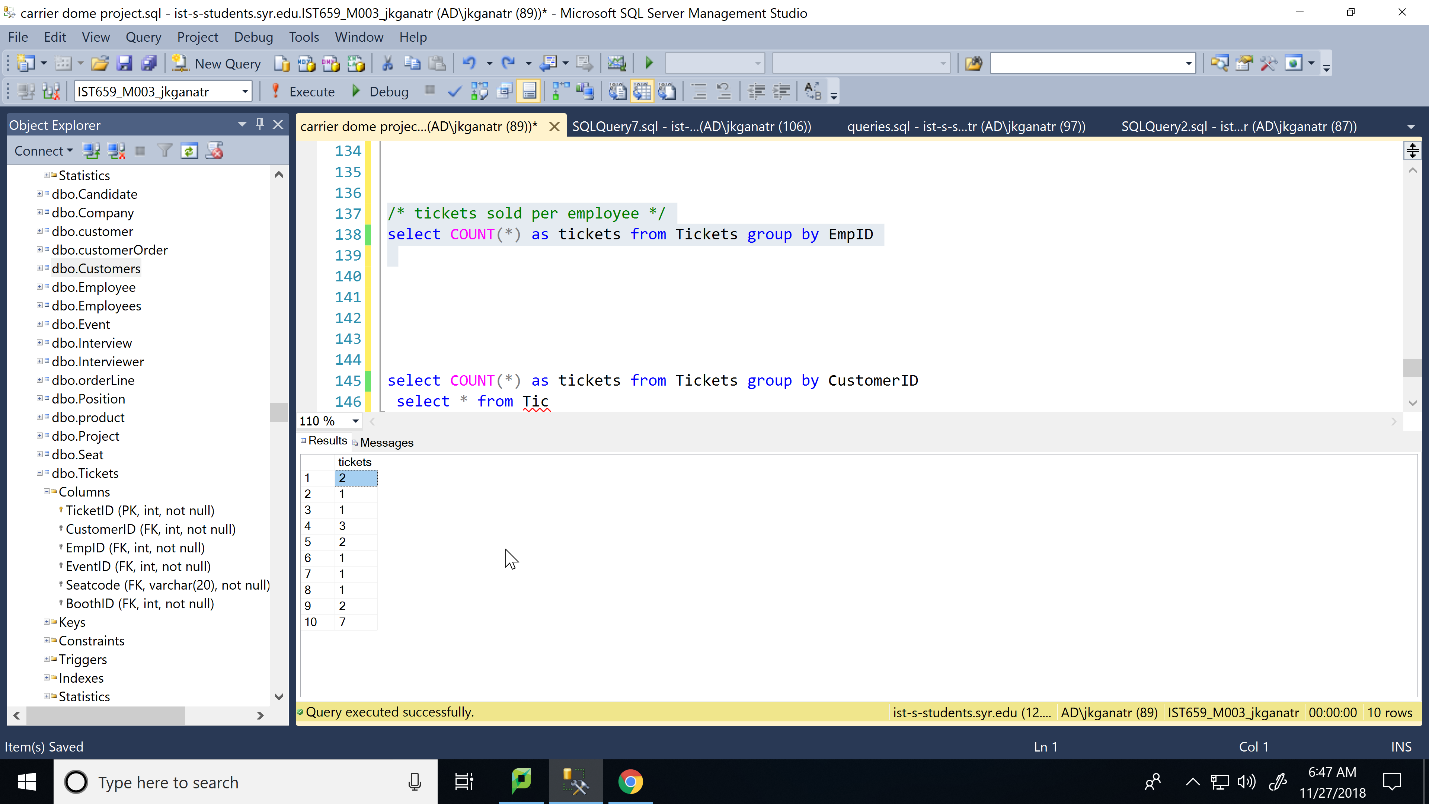


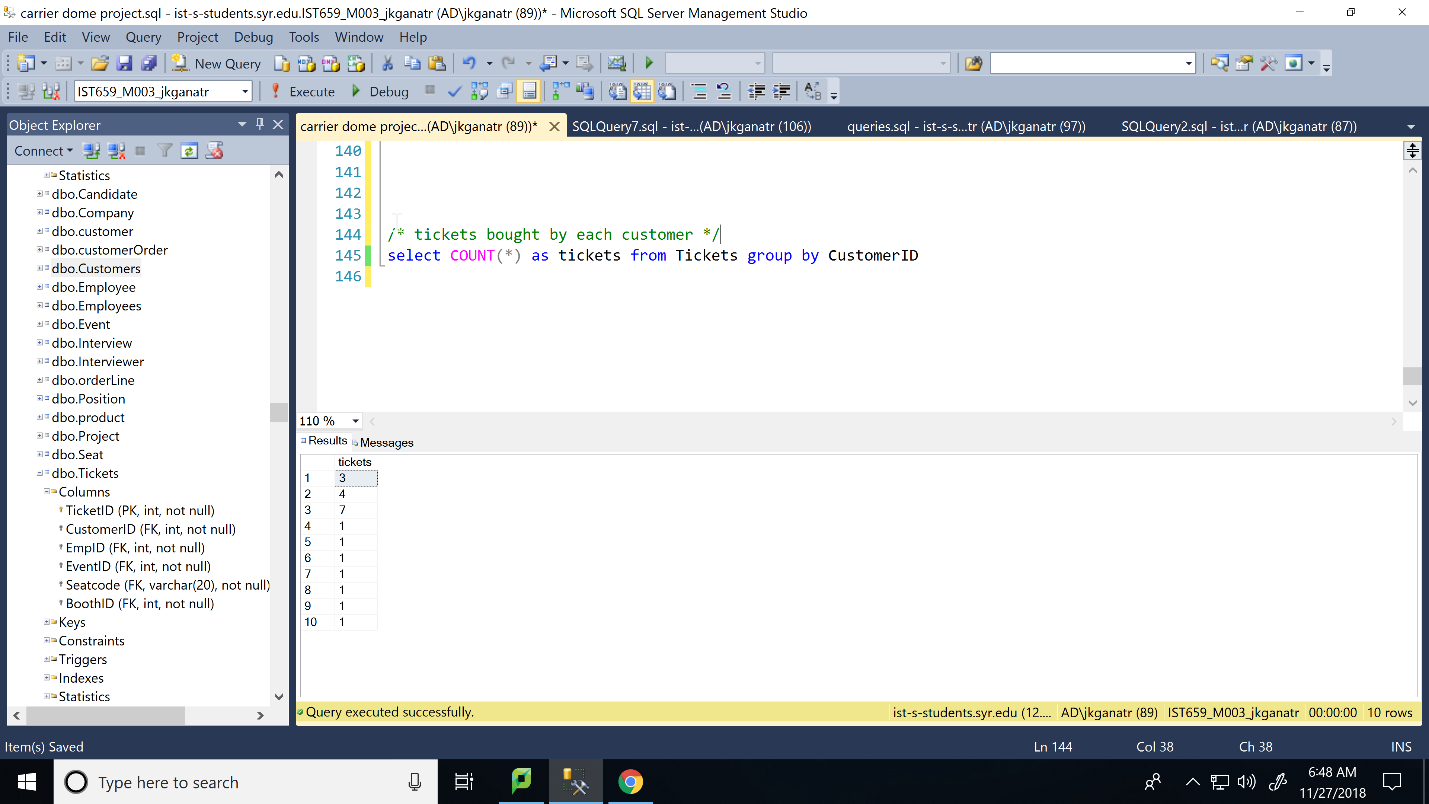
Answering Data Questions





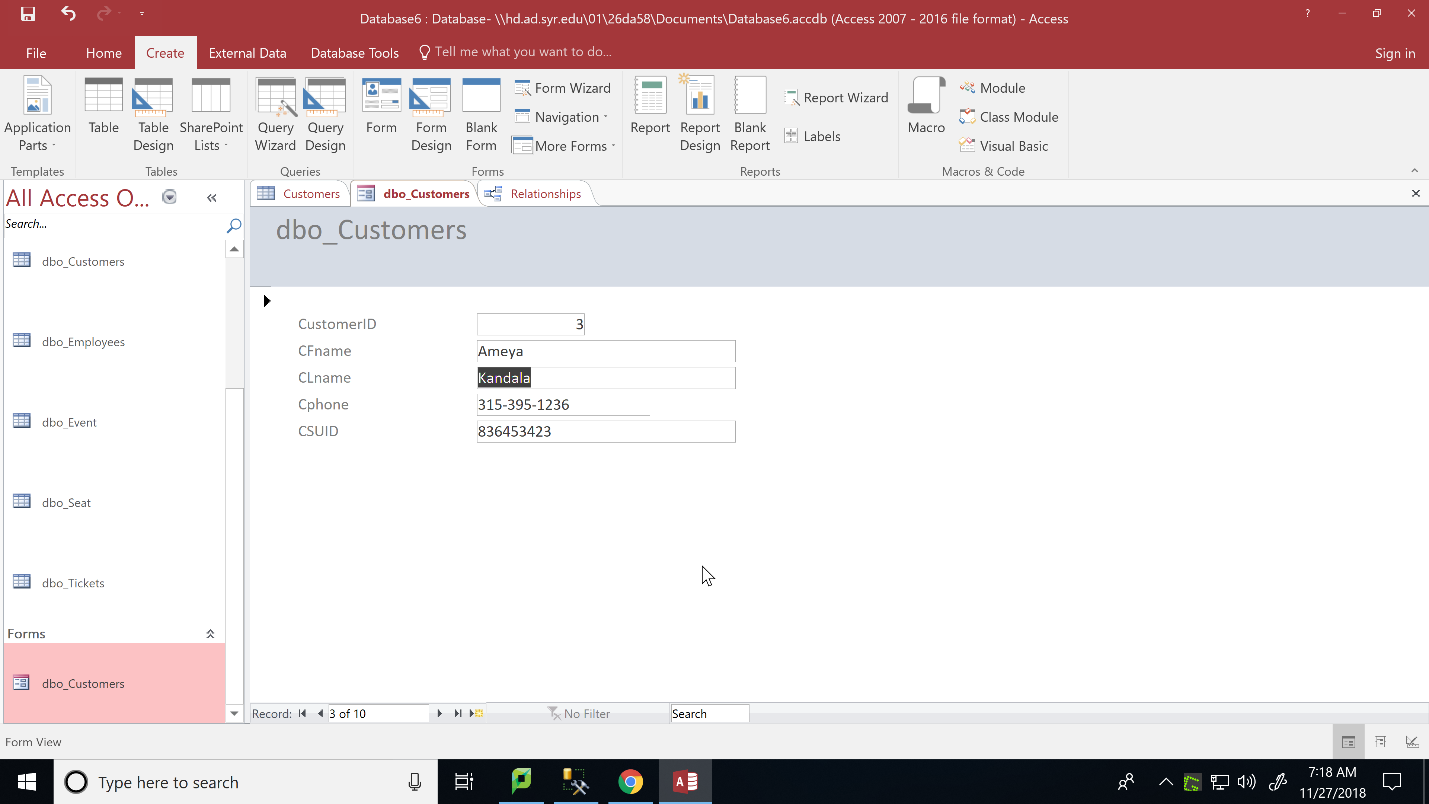




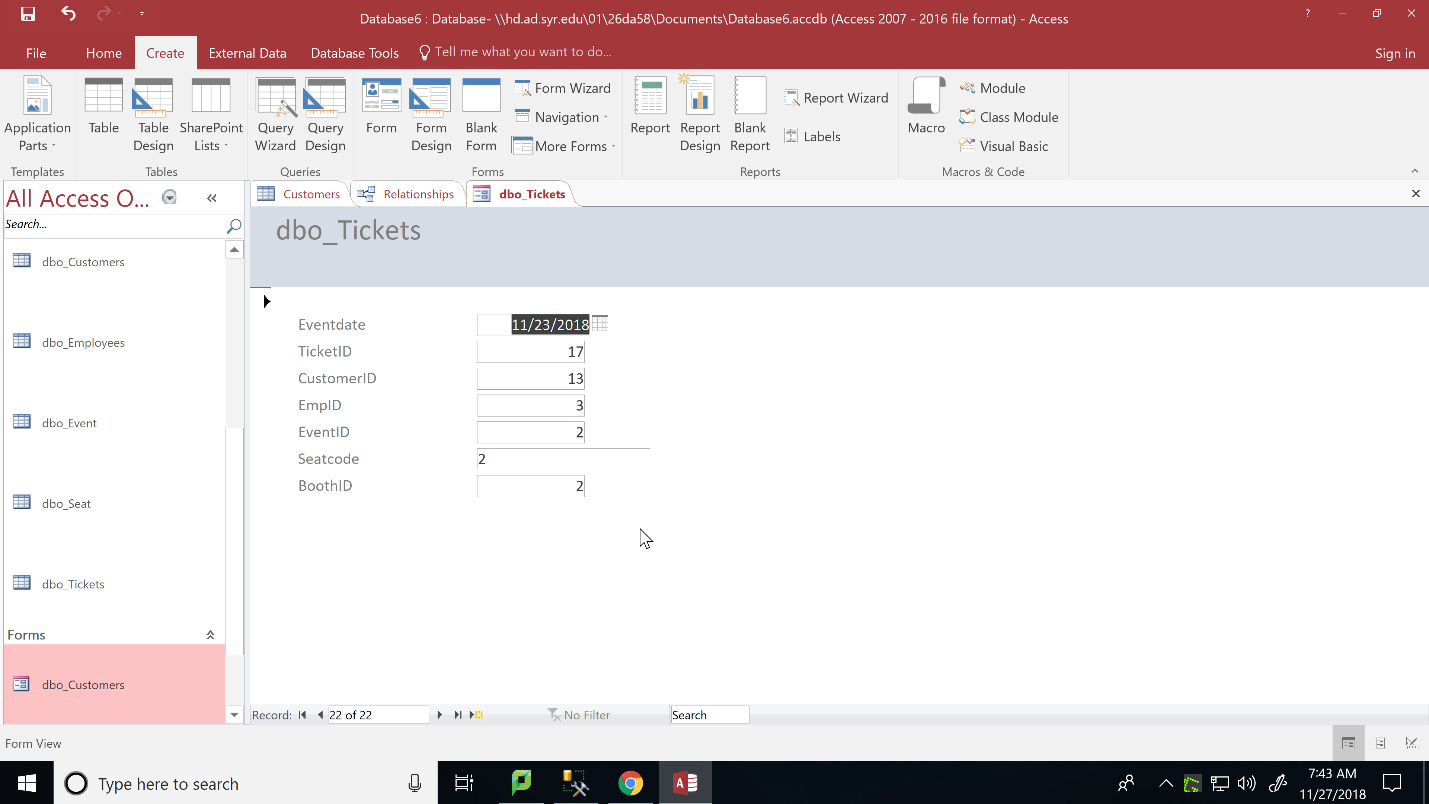


Creating forms for input of data

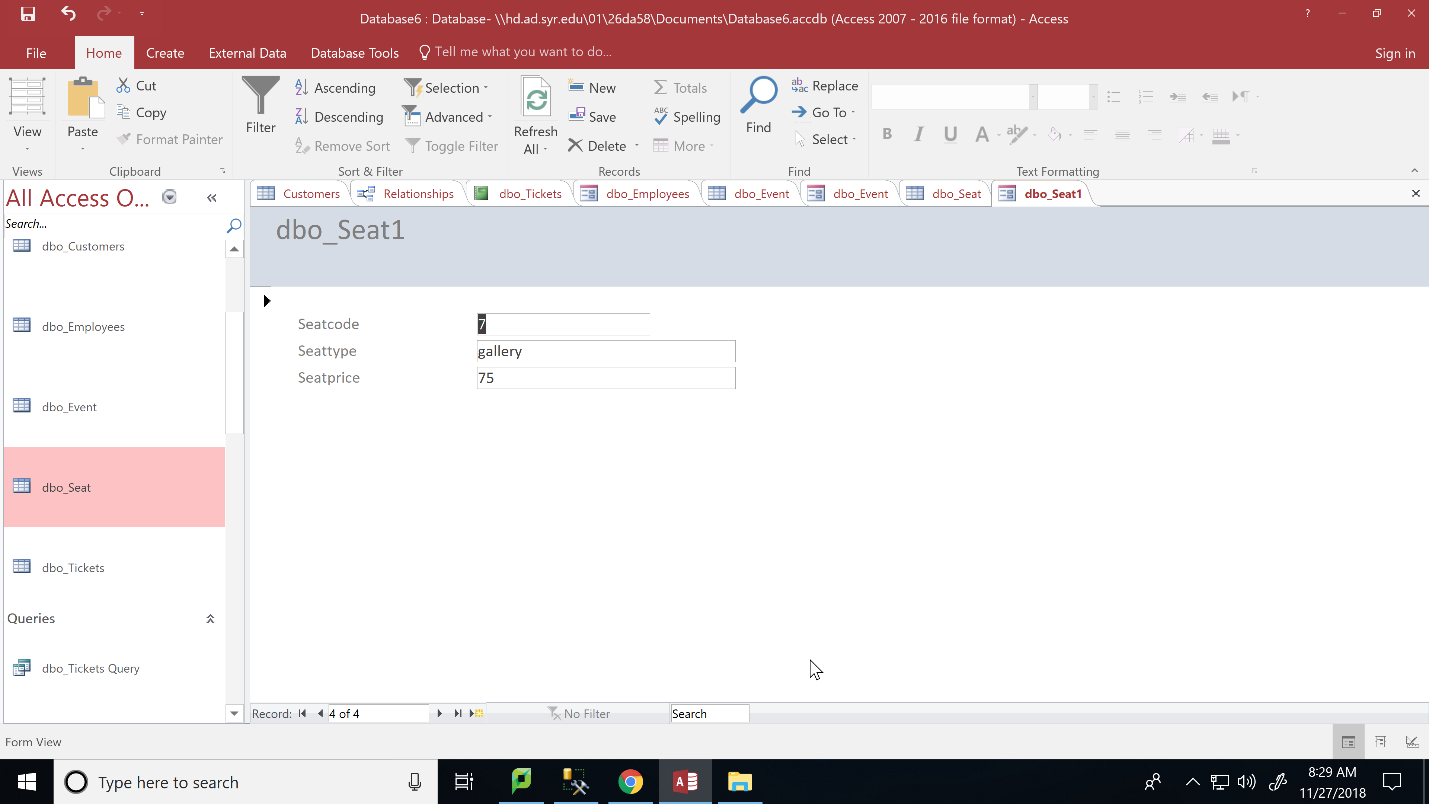
Insert into customers



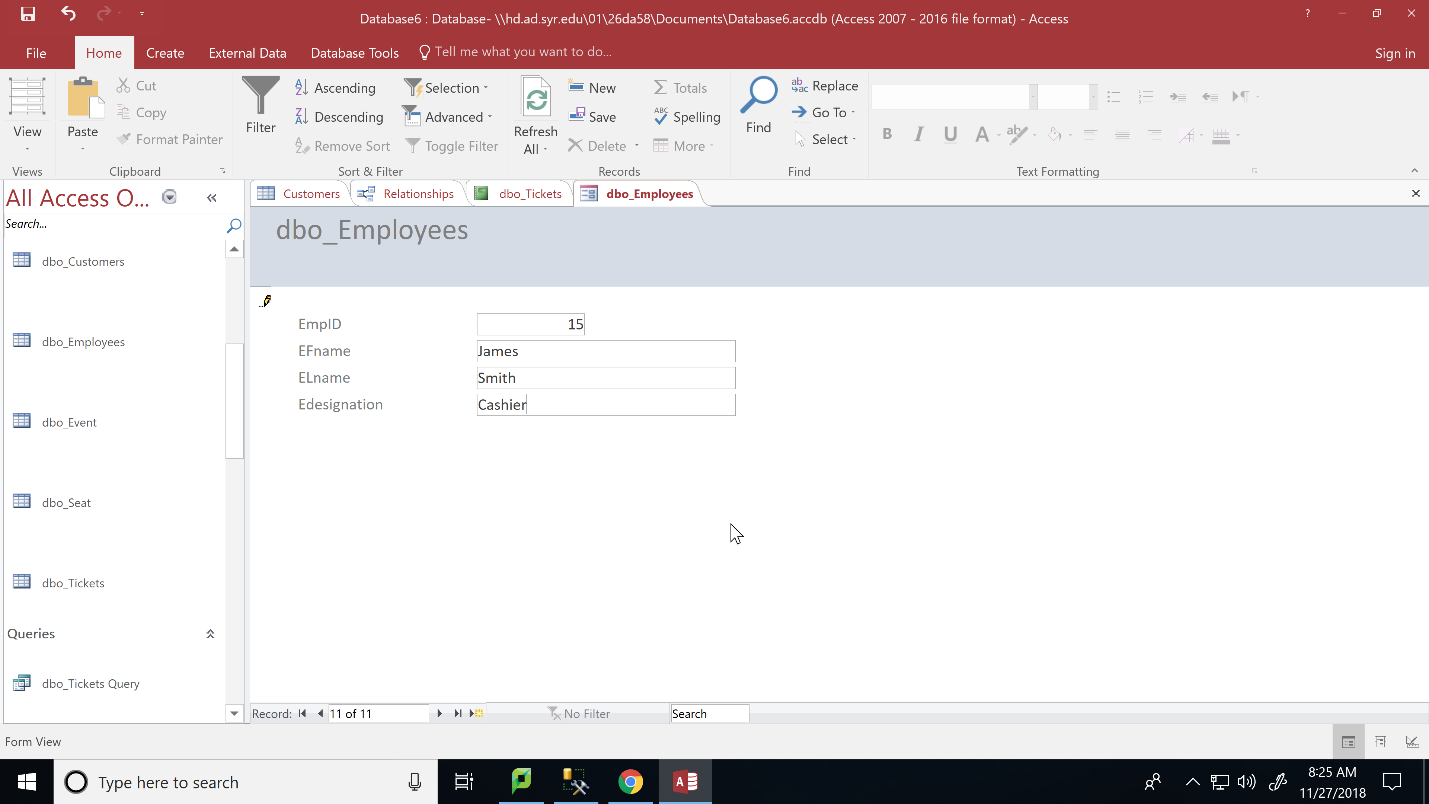
Insert into tickets

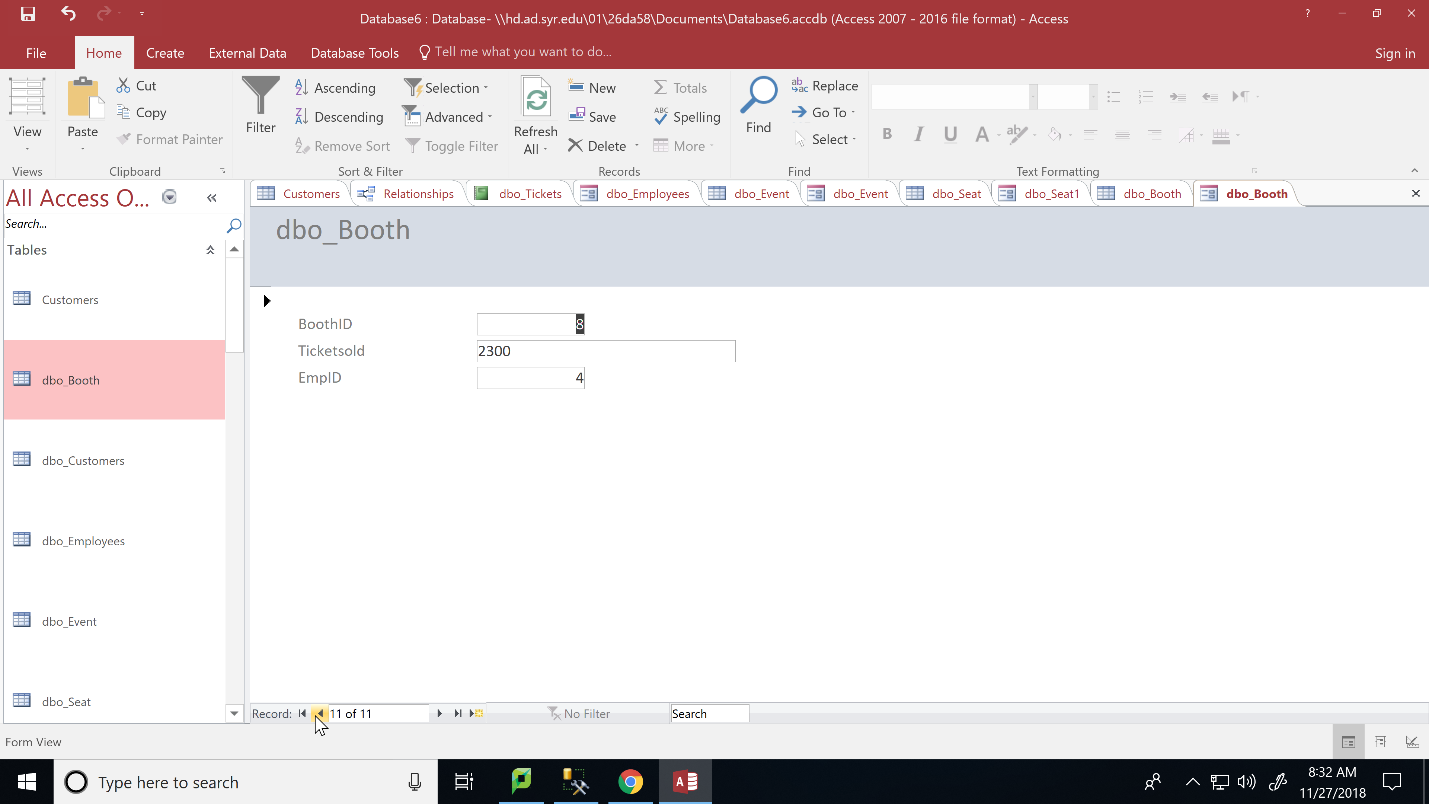


Insert into seat



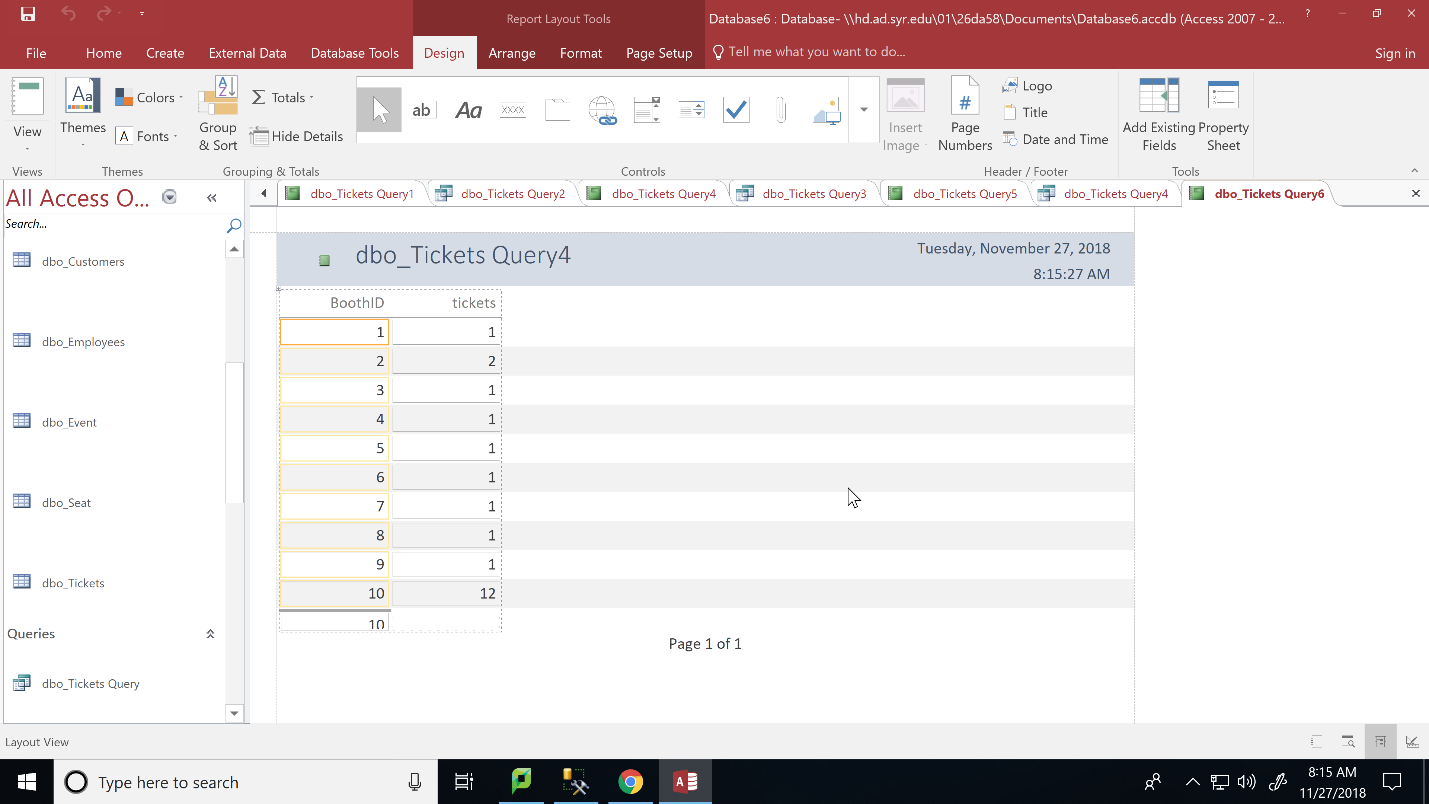
Insert into event



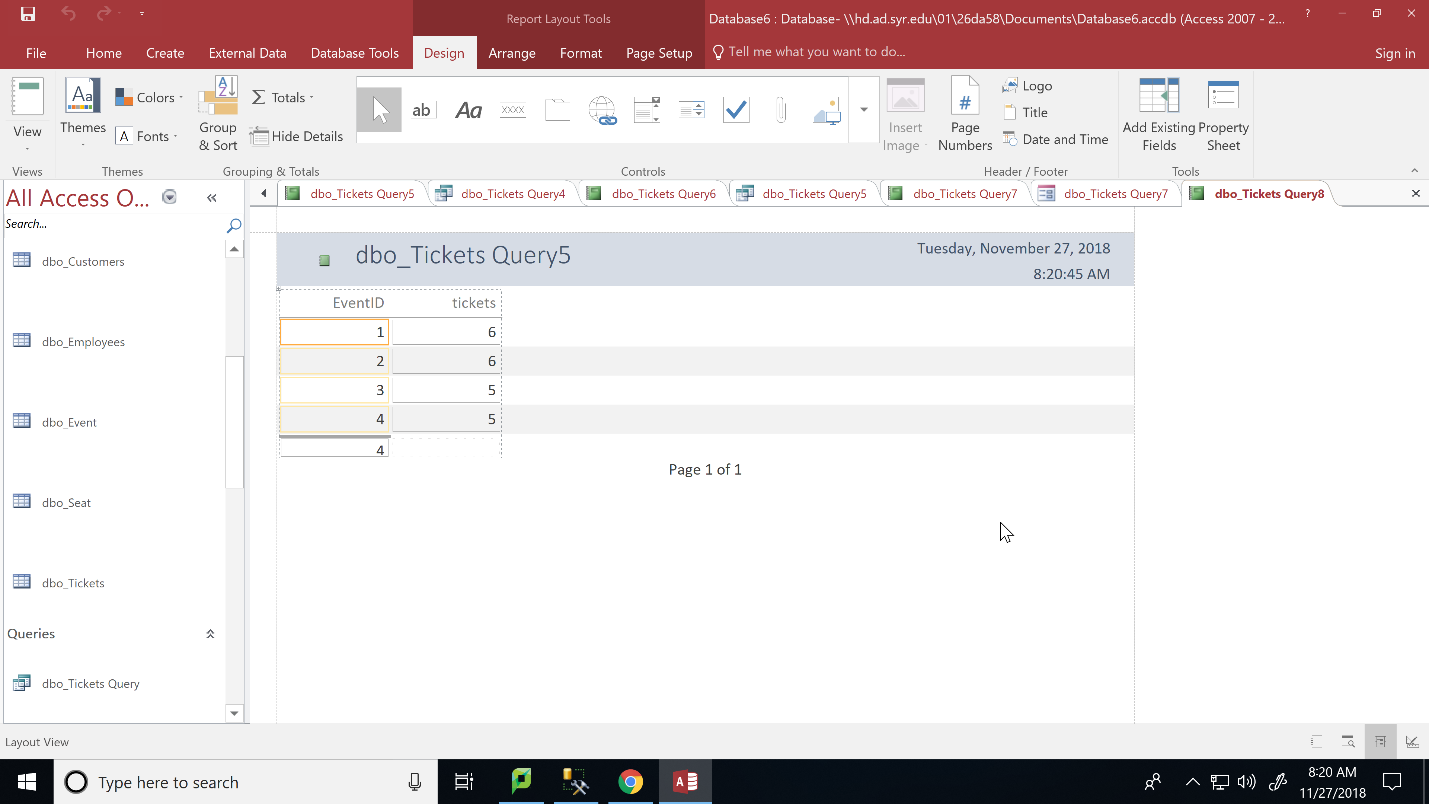
Insert into booth

Creating reports

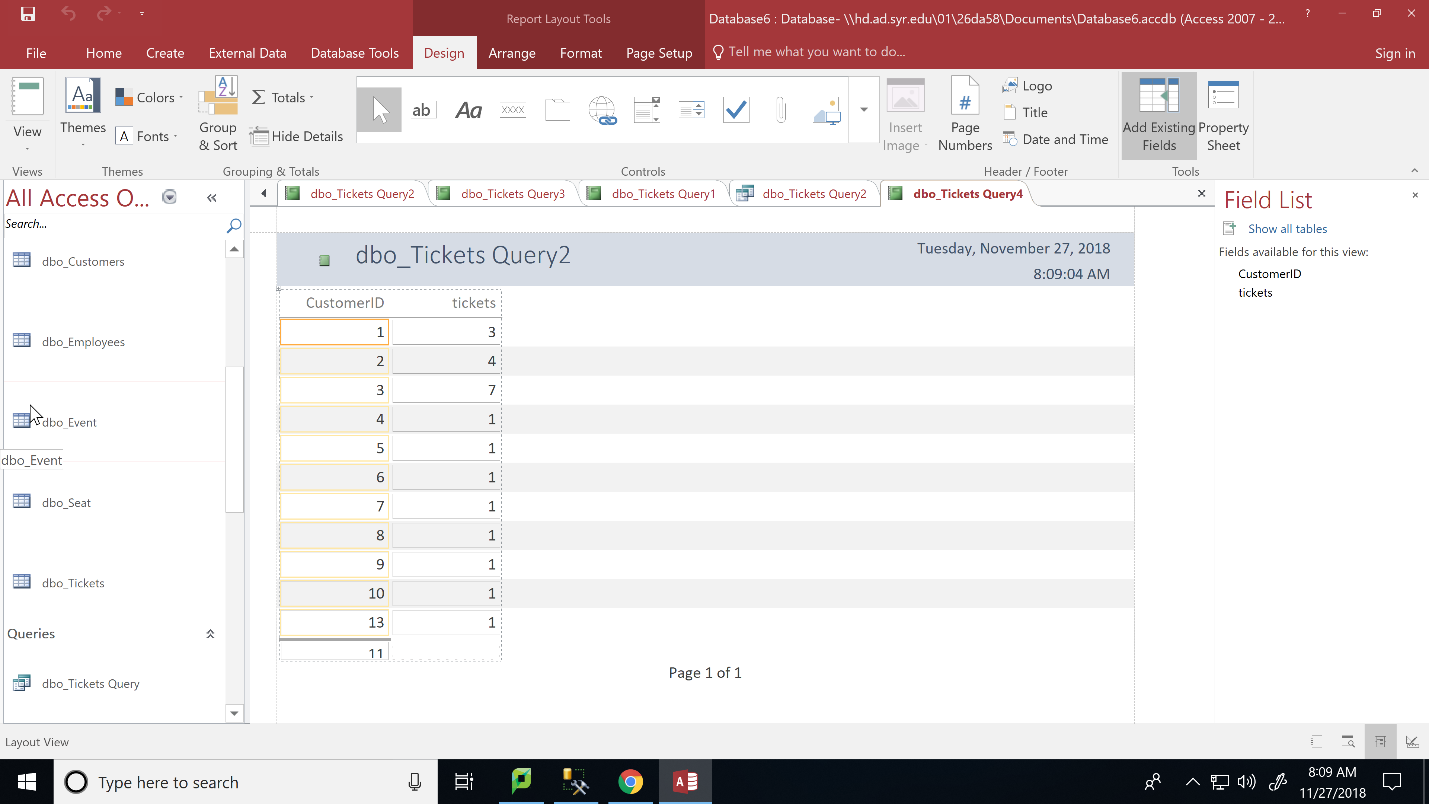
Report to show count of tickets sold per booth



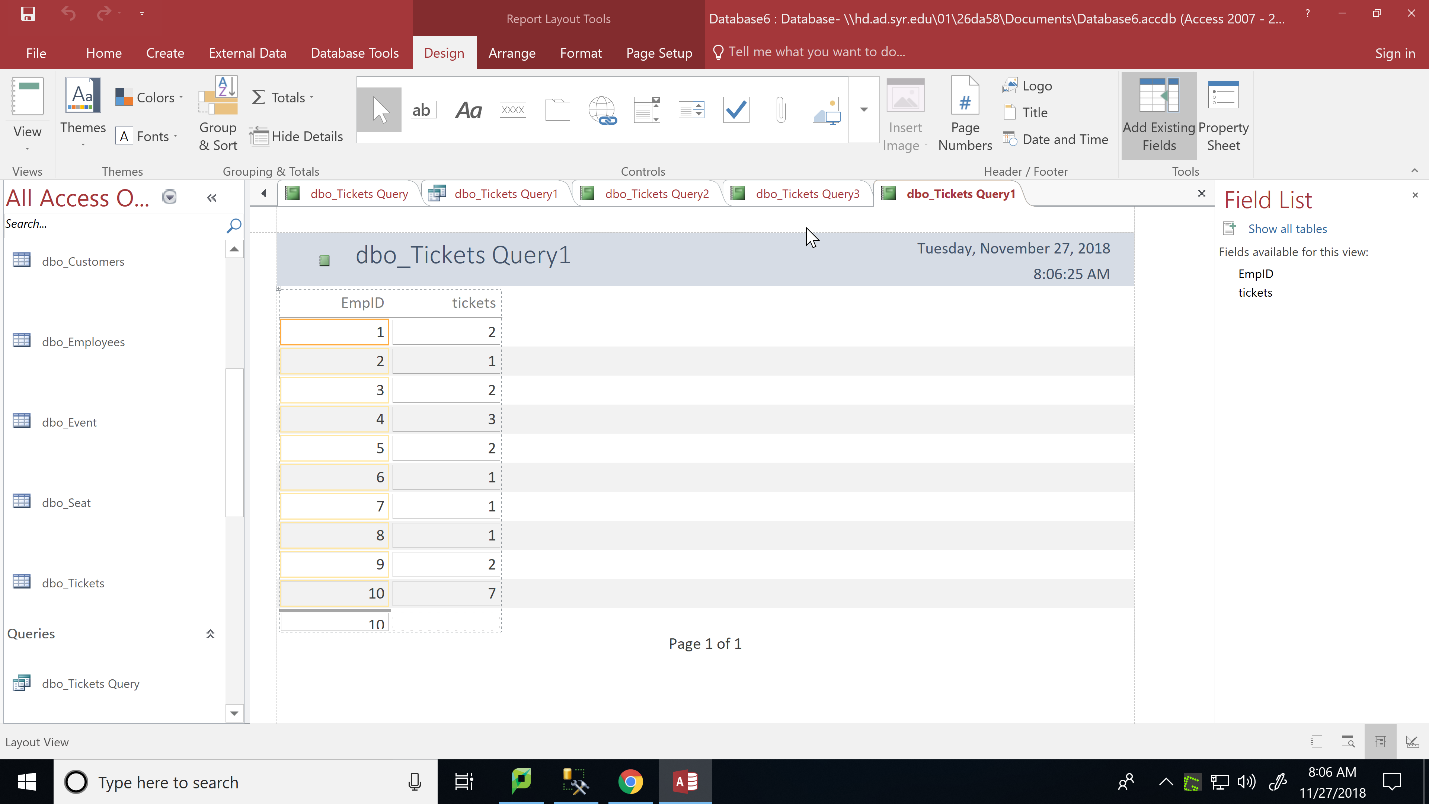
Report to show count of tickets sold for each event



Report to show number of tickets bought by each customer



Report to show number of tickets sold by each employee



Report to show number of tickets sold per seat type

