James Schnitzer

3/24/2017

1. Members: James Schnitzer
   1. All tables
2. All Queries to create and populate, attached: summerkitchen.txt
3. Queries are attached in queries.txt and pictures of results are attached in folder (pictures) below are descriptions.
   1. This query finds all employees.
   2. This query finds all stores.
   3. This query finds all employees paid above minimum wage per hour ($10).
   4. Find where each employee works is found with this query.
   5. Find which server served which customer with this query.
   6. Find which customers attended which locations with this query.
   7. This query finds which location had the most customers.
   8. Sales per employee are found with this query.
   9. This query finds the sales per store.
   10. Hours worked per employee are found with this query.
   11. Hours worked per store are found with this query.
   12. Find what each cook is to be paid with this query.
   13. What each host is to be paid is found with this query.
   14. Where each customer attended is found with this query.
   15. Which payment methods were preferred overall is shown with this query.
   16. This query shows which payment method was preferred per waiter.
   17. This query shows which payment method was preferred per store
   18. Number of customers attended per server is found with this query.
   19. Number of customers attended per store is found with this query.
   20. Find all managers with this query
   21. Find which stores managers work at with this query.
   22. Who has worked more than 5 hours is show with this query.
   23. Another version of the last query (22).
   24. Average cost per customer (per order) are shown here.
   25. Average cost per customer (per server) are shown here.
   26. Average cost per customer (per store) are shown here.
   27. CASH Sales per Server are shown here.
   28. CREDIT Sales per Server are shown here.
   29. CASH Sales per store are shown here.
   30. CREDIT Sales per store are shown here.
   31. Fire an employee (MUST ALSO DELETE THEM FROM THE IS\_A and shift TABLES FIRST) with this query.
   32. Update Tuple (CANNOT UPDATE Primary Keys this way) with this query.
   33. This query joins employee and location to find which locations servers work at.
   34. This query renames a table.
   35. UNION between which employees were hired since 2010 and work at the Chandler location are shown here.
   36. Find which employees do not work at the Cornhisker or the Chandler locations with this query.
   37. Which employees are managers and are paid above $10,000 is shown with this query. (The online database did not like my query but according to the text book it was syntactically correct.)
4. Original SQL specification form Checkpoint 1

**create table** *store\_location* (

*address* **char(**25)

*hours\_open* **char**(9)

*date\_opened* **date**(10)

**primary key**(*address*))

**create table** *employee*(

*SSN* **int(**9)

*emp\_name* **char**(18)

*date\_started* **date**(10)

*pin* **int**(4)

*wage* **float**(3,2)

**primary key**(*SSN*))

**create table** *customer* (

*customer\_num* **int(**)

*payment\_type* **char**(7)

*party\_size* **int()**

**primary key**(*customer\_num*))

**create table** *food* (

*name* **char(**25)

*type* **char**(7)

*price* **float**(5,2)

**primary key**(*name, type*))

**create table** *shift* (

*shift\_id* **int(**)

*begin* **int**()

*end* **int**()

*type* **string**(7)

*lenght* **int**()

*date* **date**(10)

**primary key**(*shift\_id*))

**create table** *order* (

*order\_num* **int(**)

*time\_ordered* **string**(4)

*table\_num* **int**()

**primary key**(*order\_num*))

**Relations**

**create table** *works\_there* (

**join** *store\_location, employee*

**primary key**(*address, SSN*))

**create table** *worked\_this* (

**join** *shift, employee*

**primary key**(*shift\_id, SSN*))

**create table** *served* (

**join** *customer, employee*

**primary key**(*customer\_num, SSN*))

**create table** *bought* (

**join** *customer, order*

**primary key**(*customer\_num, order\_num*))

**create table** *on\_order* (

**join** *food, order*

**primary key**(*name, type, order\_num*))

1. Differences
   1. Dropped the ‘food’ table
      1. Being able to add food to the same order was not going to work because there would be a possibility of having two of the same items on the same order. This was dropped instead of reworking the database to allow for multiple of the same item to be added to an order. That is also why there is a new attribute, bill\_ total for the customer table.
   2. Dropped the employee.wage variable and created three more specializations for the employee
      1. This allowed for diverse types of employee to be specialized based on their role. For example, the is\_waiter is directly connected to the customers because they are the only ones in contact with selling items to the customers. Wage was dropped because of these being created.
   3. There is an extra table “adjusted\_length” that was created.
      1. The online creator would not let me create the derived variable “length” without making a special view of the table so that was made and used in the queries.
   4. Tables altered for primary and foreign keys.
      1. When I made the tables I did not see where to set a primary key so those and the foreign keys were added later in the designing process.
   5. On the shift table the date attribute are all the same, that is because that is automatically generated when the entry is made, so in an application use it will be dependent on when the employee clocks in.
   6. Relations are not tables, when using the online maker to create relations it just connected them with keys instead of making tables as shown previously.
   7. SSN was added to the Shift and Customer tables to allow for direct connectivity between the shift/customer and the employee. This allowed for multiple queries to be made possible. For the same reason primary\_store was added to the employee table for the relation between employee and store.