CECS 323
Introduction to Databases

Project 01 – Pizza Time

Team Members: Christopher Shih – Thanh Nguyen

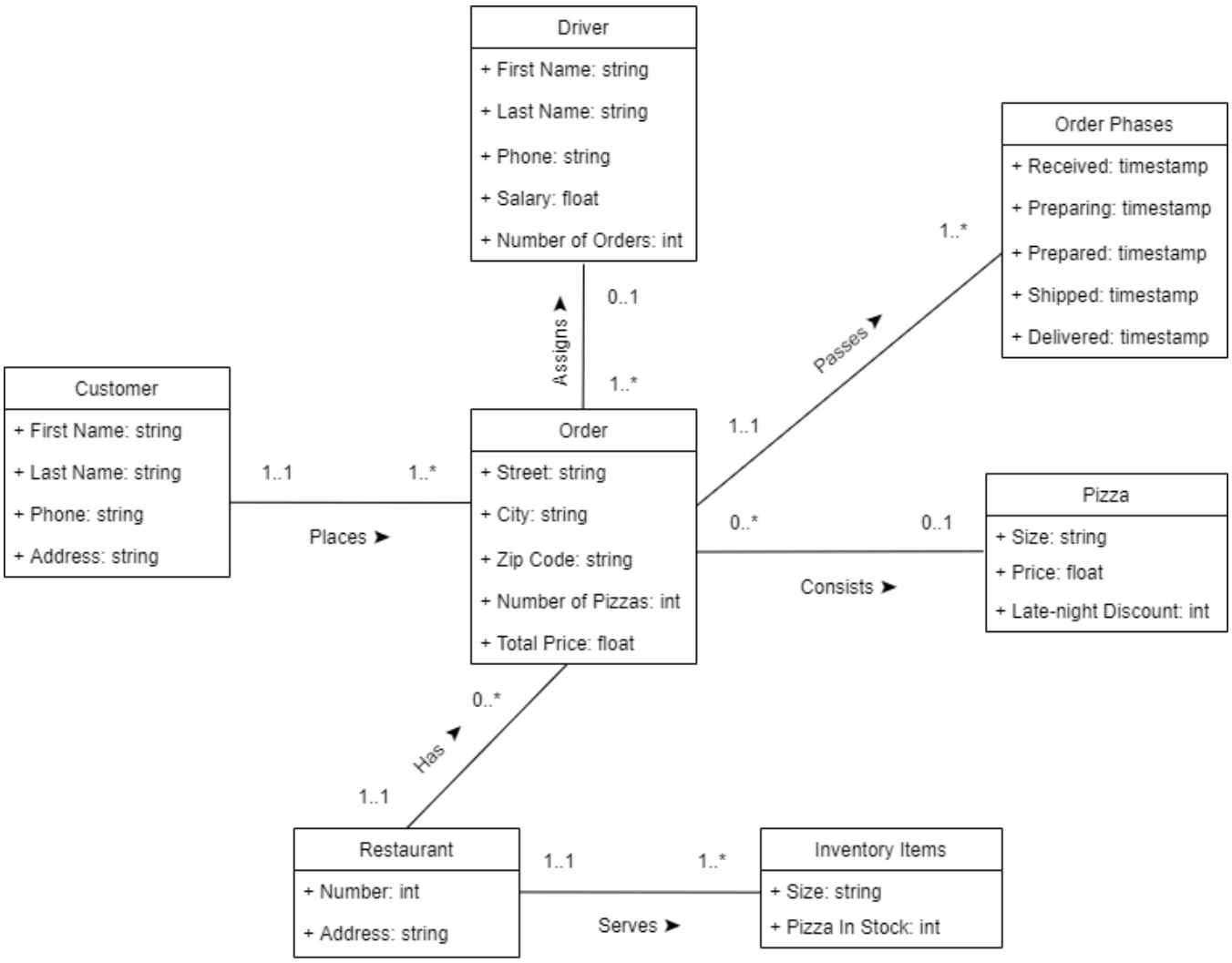
Due date: September 23 at 2:00pm

• • •

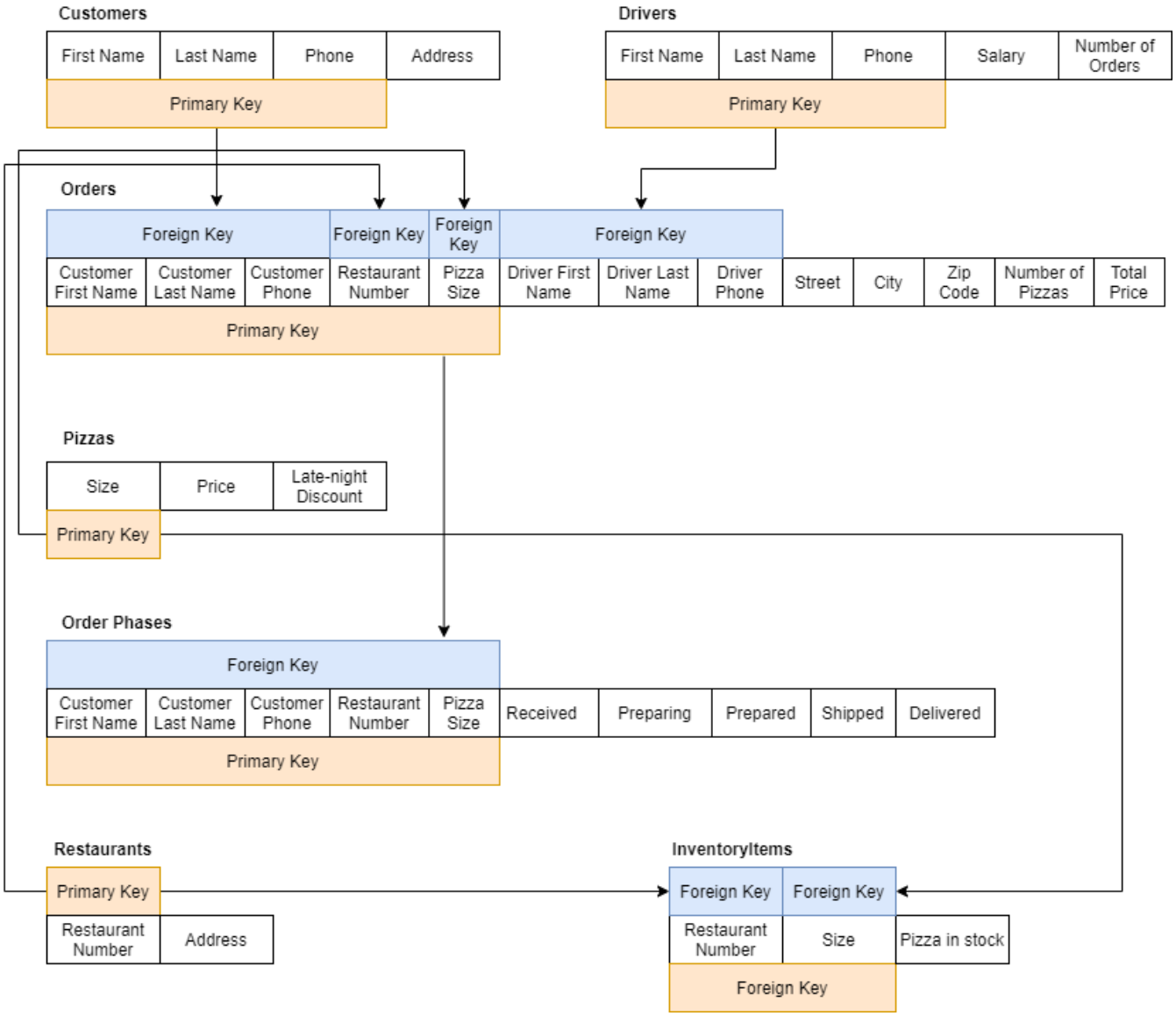
I. Class descriptions:

- **Customer** is an individual who places an order of cheese pizza from a restaurant. The restaurant records the customer's name and phone number.
- **Pizza** is a dish in a restaurant consisting of a layer of dough crust and cheese. The pizza has different sizes, and the price for each pizza is based on its size. In addition, there is a late-night discount for customer.
- **Order** is a request for food in a particular restaurant, which consists of cheese pizza in varying sizes. The order also handles the status of the pizzas. When the pizzas are ready to deliver, a driver is needed to deliver the food to customer's location.
- **Order Phase** is a distinct period in a series of a process of completing the order. One order will go through different phases.
- **Restaurant** is a place where customers come and buy food, or have it delivered to their location. A particular restaurant will hold many orders from customers, and each one will serve a number of pizzas for each size.
- **Inventory Item** is the items which are in stock will be served in each restaurant and are separated by pizza size.
- **Driver** is a person who is assigned by a restaurant to transport the foods in an order to customers. Their salaries is calculated by hourly wage.

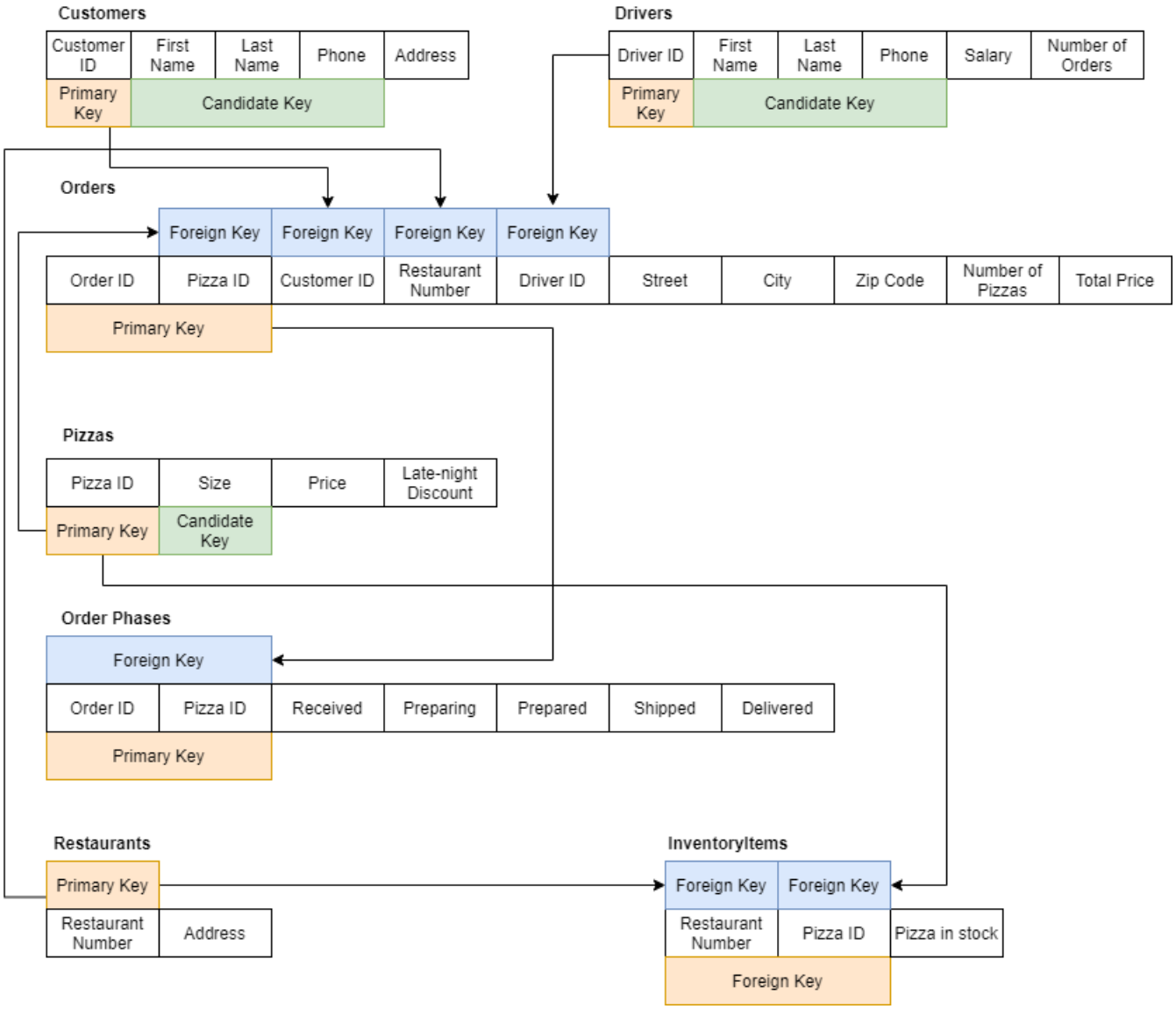
II. UML diagram:



III. First relation scheme diagram without surrogate keys:



IV. Improved relation scheme diagram with surrogate keys:



V. DDL commands to generate your tables in DataGrip/Derby:

1. CUSTOMERS table

```
create table CUSTOMERS
(
    CUSTOMER_ID INTEGER     not null
        constraint CUSTOMERS_PK
```

```
        primary key,
    FIRSTNAME   VARCHAR(20) not null,
    LASTNAME    VARCHAR(20) not null,
    PHONE       CHAR(10)    not null,
    ADDRESS     VARCHAR(50) not null
    );
```

2. DRIVERS table

```
 create table DRIVERS
 (
    DRIVER_ID       INTEGER     not null
       constraint DRIVERS_PK
          primary key,
    DRIVER_FIRSTNAME VARCHAR(20) not null,
    DRIVER_LASTNAME  VARCHAR(20) not null,
    DRIVER_PHONE     VARCHAR(10) not null,
    SALARY           DOUBLE      not null,
    NUMBER_OF_ORDERS INTEGER     not null
    );
```

3. PIZZAS table

```
 create table PIZZAS
 (
    PIZZA_ID            INTEGER     not null
       constraint PIZZAS_PK
          primary key,
    SIZE                VARCHAR(10) not null,
```

```
    PRICE               DOUBLE      not null,
    "LATE-NIGHT_DISCOUNT" INTEGER     not null
    );
```

4. RESTAURANTS table

```
 create table RESTAURANT
 (
    RESTAURANT_NUMBER INTEGER     not null
       constraint RESTAURANT_PK
          primary key,
    ADDRESS           VARCHAR(50) not null
    );
```

5. INVENTORYITEMS table

```
 create table INVENTORYITEMS
 (
    RESTAURANT_NUMBER INTEGER not null
       constraint INVENTORYITEMS_RESTAURANT_RESTAURANT_NUMBER_FK
          references RESTAURANT,
    PIZZA_ID          INTEGER not null
       constraint INVENTORYITEMS_PIZZAS_PIZZA_ID_FK
          references PIZZAS,
    PIZZA_IN_STOCK    INTEGER not null,
    constraint INVENTORYITEMS_PK
       primary key (RESTAURANT_NUMBER, PIZZA_ID)
    );
```

6. ORDERS table

```
create table ORDERS
(
   ORDER_ID         VARCHAR(10) not null,
   PIZZA_ID         INTEGER     not null
      constraint ORDERS_PIZZAS_PIZZA_ID_FK
         references PIZZAS,
   CUSTOMER_ID      INTEGER     not null
      constraint ORDERS_CUSTOMERS_CUSTOMER_ID_FK
         references CUSTOMERS,
   RESTAURANT_NUMBER INTEGER    not null
      constraint ORDERS_RESTAURANT_RESTAURANT_NUMBER_FK
         references RESTAURANT,
   DRIVER_ID        INTEGER     not null
      constraint ORDERS_DRIVERS_DRIVER_ID_FK
         references DRIVERS,
   STREET           VARCHAR(20) not null,
   CITY             VARCHAR(20) not null,
   ZIPCODE          CHAR(5)     not null,
   NUMBER_OF_PIZZAS  INTEGER    not null,
   TOTAL_PRICE      DOUBLE      not null,
   constraint ORDERS_PK
      primary key (ORDER_ID, PIZZA_ID)
   );
```

7. ORDERPHASES table

```
create table ORDERPHASES
```

```
(
    ORDER_ID  VARCHAR(10) not null,
    PIZZA_ID  INTEGER     not null,
    RECEIVED  TIMESTAMP   not null,
    PREPARING TIMESTAMP,
    PREPARED  TIMESTAMP,
    SHIPPED   TIMESTAMP,
    DELIVERED TIMESTAMP,
    constraint ORDERPHASES_PK
       primary key (ORDER_ID, PIZZA_ID),
    constraint ORDERPHASES_ORDERS_ORDER_ID_PIZZA_ID_FK
       foreign key (ORDER_ID, PIZZA_ID) references ORDERS
    );
```

VI. Your numbered answers to the required SQL SELECT statements:

1. Select all the "large" pizzas that were ordered

```
SELECT ORDER_ID, FIRSTNAME, LASTNAME, RESTAURANT_NUMBER, SIZE,
PRICE, NUMBER_OF_PIZZAS, TOTAL_PRICE, DRIVER_ID, STREET, CITY,
ZIPCODE

FROM CUSTOMERS C

   INNER JOIN ORDERS O ON C.CUSTOMER_ID = O.CUSTOMER_ID

   INNER JOIN PIZZAS P ON O.PIZZA_ID = P.PIZZA_ID

WHERE P.SIZE = 'Large';
```

	ORDER_ID	FIRST...	LAST...	RESTAURANT_N... 1	SIZE	PRICE	NUMBER_OF_PIZZAS	TOTAL...	DRIVE...	STREET	CITY	ZIPCO
1	01_01	Thanh	Nguyen	1	Large	10.99	3	32.97	1	10282 Melric Ave	Westminster	92683
2	02_04	Emma	Lee	2	Large	10.99	3	32.97	2	12456 Loma Ave	Long Beach	90814
3	03_02	Emma	Lee	3	Large	10.99	1	10.99	5	4578 Loma Ave	Long Beach	90814

2. Select all orders placed on June 30, 2004

```
SELECT *

FROM ORDERS O INNER JOIN ORDERPHASES P

    ON O.ORDER_ID = P.ORDER_ID AND O.PIZZA_ID = P.PIZZA_ID

WHERE DATE(P.RECEIVED) = '2004-06-30';
```

Example Output:

```
01_01,1,1,1,1,10282 Melric Ave,Westminster,92683,2,15.98,01_01,1,2004-06-30
10:07:58.000000000,2004-06-30 10:10:58.000000000,2004-06-30
10:20:25.000000000,2004-06-30 10:30:36.000000000,2004-06-30 10:55:23.000000000
01_01,3,1,1,1,10282 Melric Ave,Westminster,92683,3,32.97,01_01,3,2004-06-30
10:07:58.000000000,2004-06-30 10:15:00.000000000,2004-06-30
10:28:21.000000000,2004-06-30 10:30:36.000000000,2004-06-30 10:55:23.000000000
02_02,1,4,2,2,1247 West Orange Ave,Anaheim,92804,3,23.97,02_02,1,2004-06-30
16:24:02.000000000,2004-06-30 16:25:16.000000000,2004-06-30
16:35:14.000000000,2004-06-30 16:43:00.000000000,2004-06-30 16:57:31.000000000
02_02,4,4,2,2,1247 West Orange Ave,Anaheim,92804,1,12.99,02_02,4,2004-06-30
16:24:02.000000000,2004-06-30 16:30:25.000000000,2004-06-30
16:24:02.000000000,2004-06-30 16:43:00.000000000,2004-06-30 16:57:31.000000000
03_01,4,2,3,4,10235 East 4th Str,Long Beach,90814,1,12.99,03_01,4,2004-06-30
12:05:02.000000000,2004-06-30 12:10:15.000000000,2004-06-30
12:25:31.000000000,2004-06-30 12:28:02.000000000,2004-06-30 12:52:07.000000000
```

3. Determine the exact time that an order of your choosing was marked as Delivered

```
SELECT DISTINCT O.ORDER_ID, FIRSTNAME, LASTNAME, STREET, CITY,
ZIPCODE, P.DELIVERED

FROM ORDERS O
```

INNER JOIN ORDERPHASES P ON O.ORDER_ID = P.ORDER_ID AND O.PIZZA_ID = P.PIZZA_ID

INNER JOIN CUSTOMERS C on O.CUSTOMER_ID = C.CUSTOMER_ID;

ORDER_ID	FIRSTNAME	LASTNAME	STREET	CITY	ZIPCODE	DELIVERED
01_04	Thanh	Nguyen	4578 Buena Park	Los Angeles	92007	2021-06-02 20:57:45.000000000
03_02	Emma	Lee	4578 Loma Ave	Long Beach	90814	2019-10-14 20:59:01.000000000
02_04	Emma	Lee	12456 Loma Ave	Long Beach	90814	2017-09-09 16:25:14.000000000
01_03	Thanh	Nguyen	1245 Nieto Ave	Long Beach	90814	2015-06-24 17:10:23.000000000
02_03	Anthony	Lim	14789 Ashdale	Irvine	92620	2012-12-24 21:12:58.000000000
01_02	Anna	Stone	1254 Thornwood Dr	San Jose	95123	2010-09-21 16:02:35.000000000
02_02	Emma	Lee	1247 West Orange Ave	Anaheim	92804	2004-06-30 16:57:31.000000000

4. Select all orders delivered to the 90814 zip code

SELECT DISTINCT O.ORDER_ID, FIRSTNAME, LASTNAME, STREET, CITY, ZIPCODE, DELIVERED

FROM ORDERS O

INNER JOIN CUSTOMERS C ON O.CUSTOMER_ID = C.CUSTOMER_ID

INNER JOIN ORDERPHASES P ON O.ORDER_ID = P.ORDER_ID and O.PIZZA_ID = P.PIZZA_ID

WHERE ZIPCODE = '90814';

ORDER_ID	FIRSTNAME	LASTNAME	STREET	CITY	ZIPCODE	DELIVERED
01_03	Thanh	Nguyen	1245 Nieto Ave	Long Beach	90814	2015-06-24 17:10:23.000000000
02_04	Emma	Lee	12456 Loma Ave	Long Beach	90814	2017-09-09 16:25:14.000000000
03_01	Anna	Stone	10235 East 4th Str	Long Beach	90814	2004-06-30 12:52:07.000000000
03_02	Emma	Lee	4578 Loma Ave	Long Beach	90814	2019-10-14 20:59:01.000000000

5. Determine all pizza sizes that are out of stock at a specific restaurant of your choosing

SELECT R.RESTAURANT_NUMBER, ADDRESS, SIZE

FROM RESTAURANTS R

• • •

```
    INNER JOIN INVENTORYITEMS I ON R.RESTAURANT_NUMBER =
I.RESTAURANT_NUMBER

    INNER JOIN PIZZAS P ON I.PIZZA_ID = P.PIZZA_ID

WHERE PIZZA_IN_STOCK = 0;
```

RESTAURANT_NUMBER	ADDRESS	SIZE
1	4897 Euclid St, Anaheim CA 92801	X-Large
2	10547 Katella Ave, Garden Grove CA 92804	Large
3	8745 La Palma Ave, Buena Park CA 90620	Medium