University of Central Florida
CGS 2545
Database Concepts

DEPARTMENT OF ELECTRICAL ENGINEERING & COMPUTER SCIENCE

COMPUTER SCIENCE DIVISION

RDBMS Concepts

- Database Normalization
  - Is the process of efficiently organizing data in a database.
  - There are two reasons of this normalization process
    - Eliminating redundant data, for example, storing the same data in more than one table.
    - Ensuring data dependencies make sense.
  - Both these reasons are worthy goals as they reduce the amount of space a database consumes and ensures that data is logically stored

RDBMS Concepts

- Database Normalization
  - Normalization guidelines are divided into normal forms
  - think of a form as the format or the way a database structure is laid out.
  - The aim of normal forms is to organize the database structure, so that it complies with the rules of first normal form, then second normal form and finally the third normal form.

RDBMS Concepts

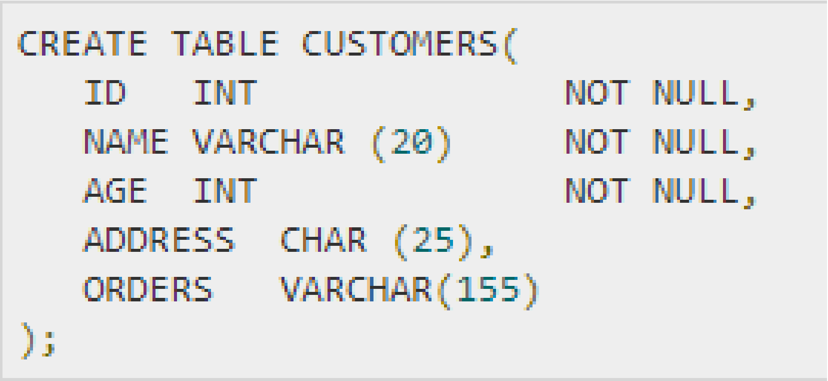
- Database Normalization
  - Normalization guidelines are divided into normal forms
  - think of a form as the format or the way a database structure is laid out.
  - The aim of normal forms is to organize the database structure, so that it complies with the rules of first normal form, then second normal form and finally the third normal form.
  - It is the choice of the designers to take it further and go to the fourth normal form, fifth normal form and so on, but in general, the third normal form is more than enough

RDBMS Concepts

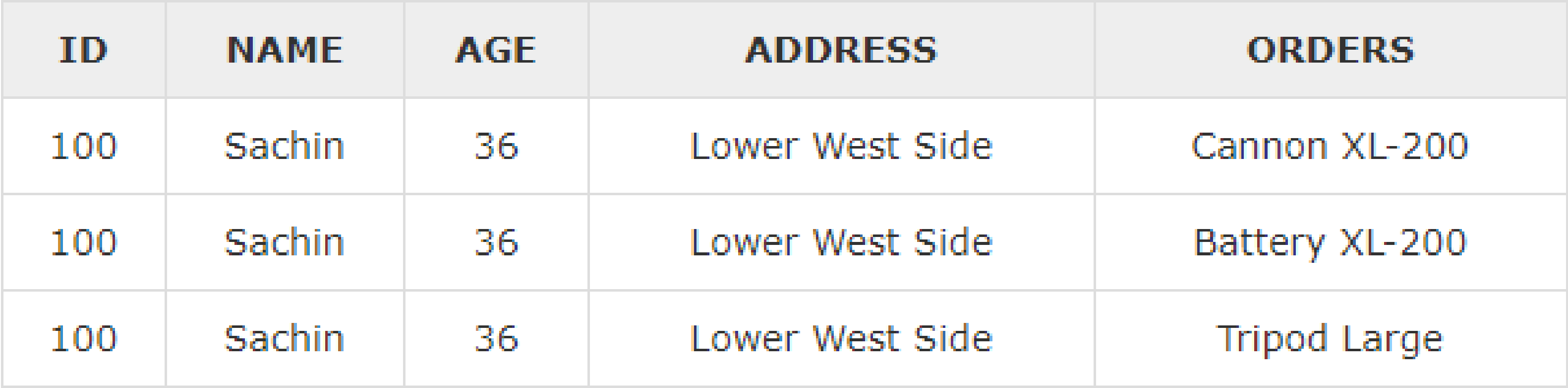
- Database Normalization
  - First Normal Form (1NF)
    - Define the data items required, because they become the columns in a table.
    - Place the related data items in a table.
    - Ensure that there are no repeating groups of data.
    - Ensure that there is a primary key.

RDBMS Concepts

- Database Normalization
  - First Normal Form (1NF) violation



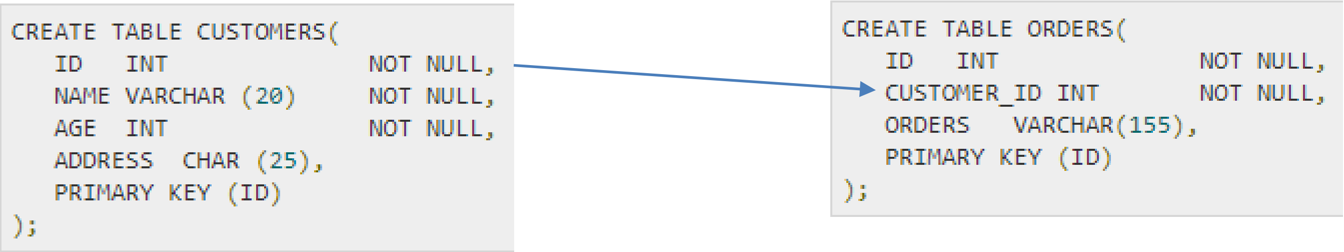
```
CREATE TABLE CUSTOMERS(
   ID   INT              NOT NULL,
   NAME VARCHAR (20)     NOT NULL,
   AGE  INT              NOT NULL,
   ADDRESS  CHAR (25),
   ORDERS   VARCHAR(155)
);
```



ID	NAME	AGE	ADDRESS	ORDERS
100	Sachin	36	Lower West Side	Cannon XL-200
100	Sachin	36	Lower West Side	Battery XL-200
100	Sachin	36	Lower West Side	Tripod Large

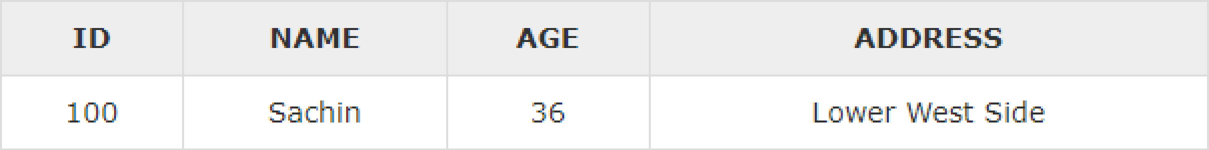
RDBMS Concepts

- Database Normalization
  - First Normal Form (1NF)

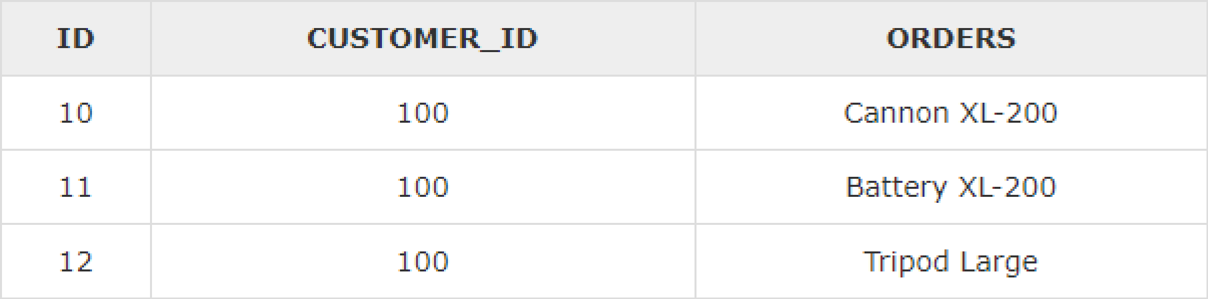


```
CREATE TABLE CUSTOMERS(
   ID   INT              NOT NULL,
   NAME VARCHAR (20)     NOT NULL,
   AGE  INT              NOT NULL,
   ADDRESS  CHAR (25),
   PRIMARY KEY (ID)
);
```

```
CREATE TABLE ORDERS(
   ID   INT              NOT NULL,
   CUSTOMER_ID INT       NOT NULL,
   ORDERS    VARCHAR(155),
   PRIMARY KEY (ID)
);
```



ID	NAME	AGE	ADDRESS
100	Sachin	36	Lower West Side



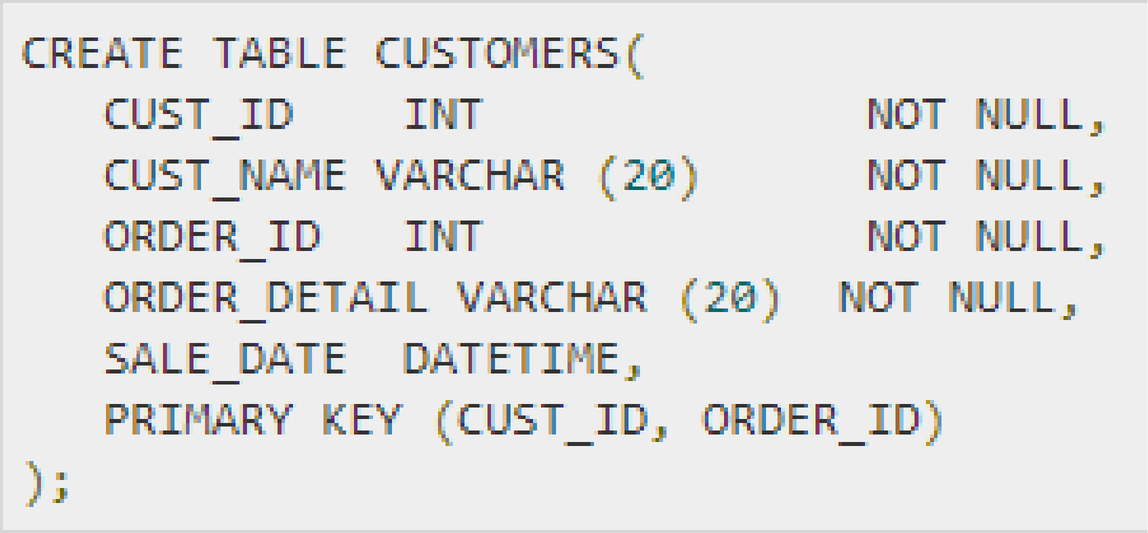
ID	CUSTOMER_ID	ORDERS
10	100	Cannon XL-200
11	100	Battery XL-200
12	100	Tripod Large

RDBMS Concepts

- Database Normalization
  - Second Normal Form (2NF)
    - The Second Normal Form states that it should meet all the rules for 1NF
    - There must be no partial dependences of any of the columns on the primary key

RDBMS Concepts

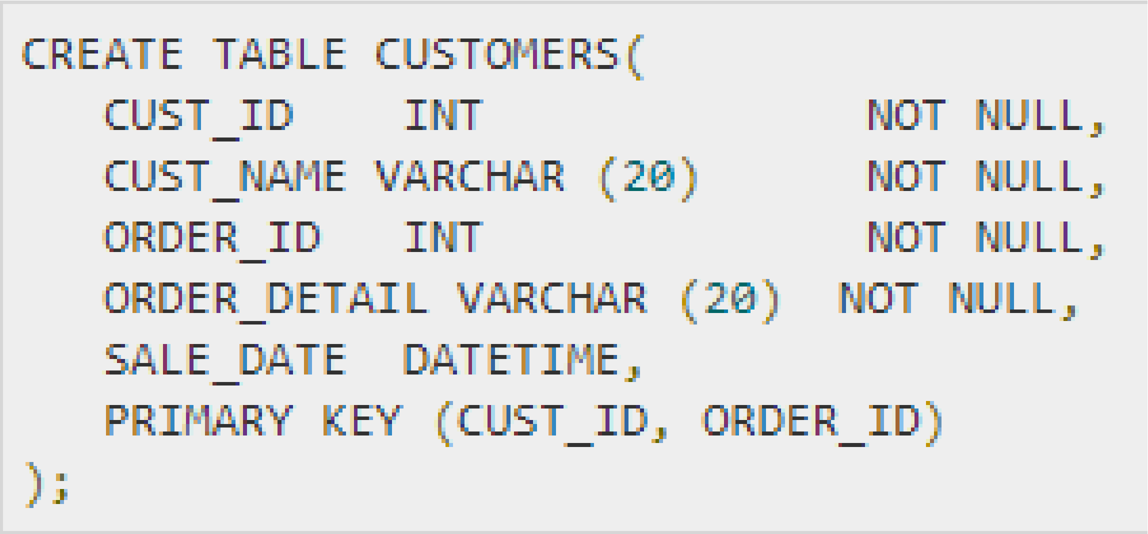
- Database Normalization
  - Second Normal Form (2NF)
    - Consider a customer-order relation and you want to store customer ID, customer name, order ID and order detail and the date of purchase



```
CREATE TABLE CUSTOMERS(
   CUST_ID    INT              NOT NULL,
   CUST_NAME VARCHAR (20)      NOT NULL,
   ORDER_ID   INT              NOT NULL,
   ORDER_DETAIL VARCHAR (20)  NOT NULL,
   SALE_DATE  DATETIME,
   PRIMARY KEY (CUST_ID, ORDER_ID)
);
```

RDBMS Concepts

- Database Normalization
  - Second Normal Form (2NF)
    - This table is in the 1NF
    - the primary key consists of the CUST_ID and the ORDER_ID; combined, they are unique assuming the same customer would hardly order the same thing



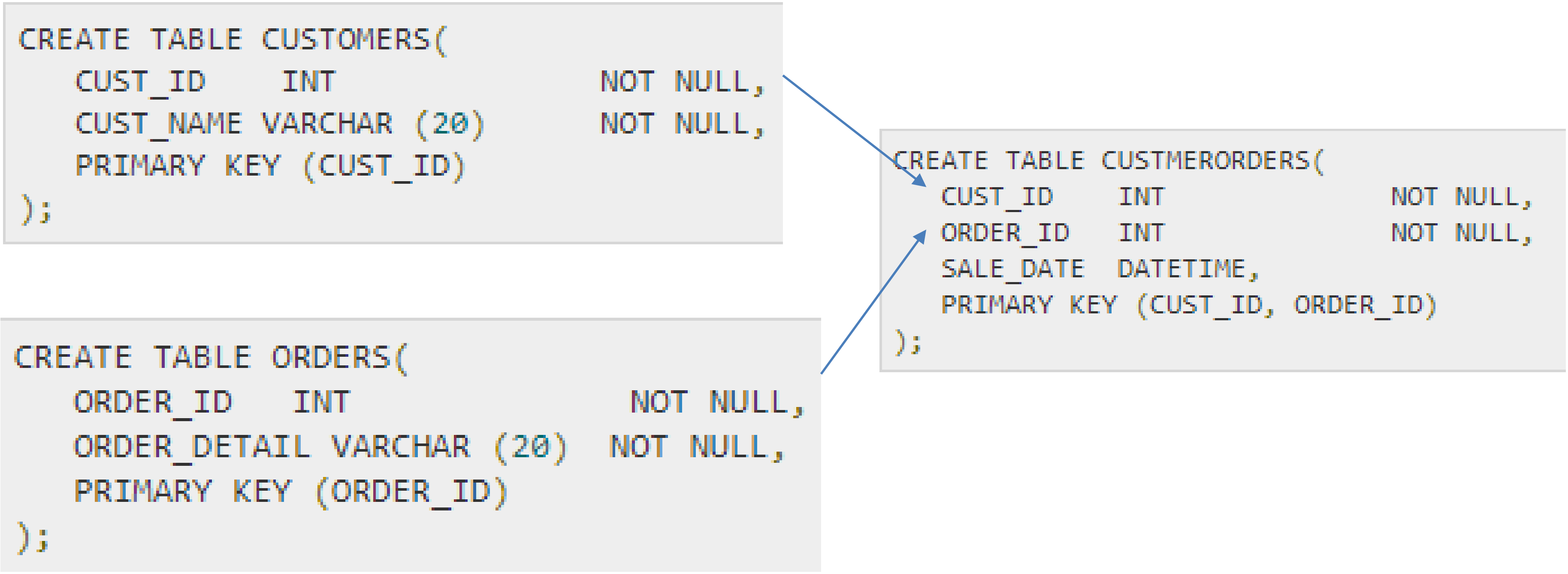
```
CREATE TABLE CUSTOMERS(
   CUST_ID    INT              NOT NULL,
   CUST_NAME VARCHAR (20)      NOT NULL,
   ORDER_ID   INT              NOT NULL,
   ORDER_DETAIL VARCHAR (20)  NOT NULL,
   SALE_DATE  DATETIME,
   PRIMARY KEY (CUST_ID, ORDER_ID)
);
```

RDBMS Concepts

- Database Normalization
  - Second Normal Form (2NF) violation
    - there are partial dependencies of primary keys and columns
      - CUST_NAME is dependent on CUST_ID
      - there's no real link between a customer's name and what he purchased
      - order detail and purchase date are dependent on the ORDER_ID
      - order detail and purchase date are not dependent on the CUST_ID,
      - there is no link between a CUST_ID and an ORDER_DETAIL or their SALE_DATE

RDBMS Concepts

- Database Normalization
  - Second Normal Form (2NF)
    - To make this table comply with the second normal form, separate the columns into three tables



```
CREATE TABLE CUSTOMERS(
   CUST_ID     INT              NOT NULL,
   CUST_NAME VARCHAR (20)       NOT NULL,
   PRIMARY KEY (CUST_ID)
);
```

```
CREATE TABLE ORDERS(
   ORDER_ID    INT                 NOT NULL,
   ORDER_DETAIL VARCHAR (20)   NOT NULL,
   PRIMARY KEY (ORDER_ID)
);
```

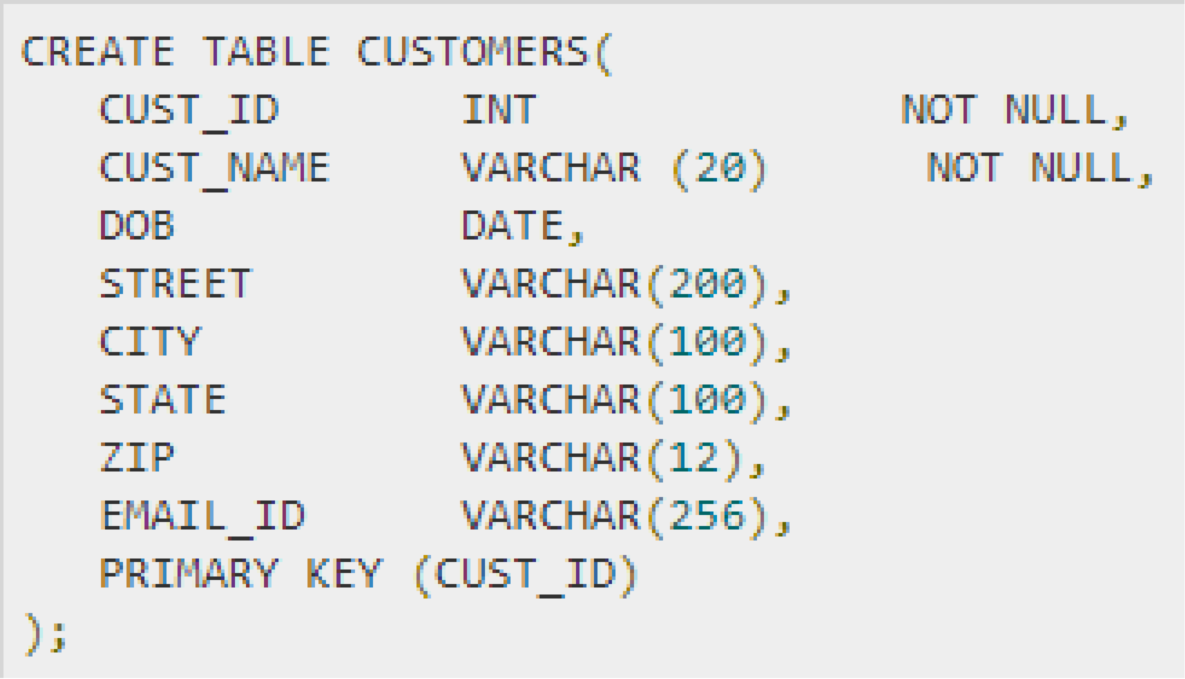
```
CREATE TABLE CUSTMERORDERS(
   CUST_ID     INT                NOT NULL,
   ORDER_ID    INT                NOT NULL,
   SALE_DATE   DATETIME,
   PRIMARY KEY (CUST_ID, ORDER_ID)
);
```

RDBMS Concepts

- Database Normalization
  - Third Normal Form (3NF)
    - The Third Normal Form states that it should meet all the rules for 2NF
    - All nonprimary fields are dependent on the primary key

RDBMS Concepts

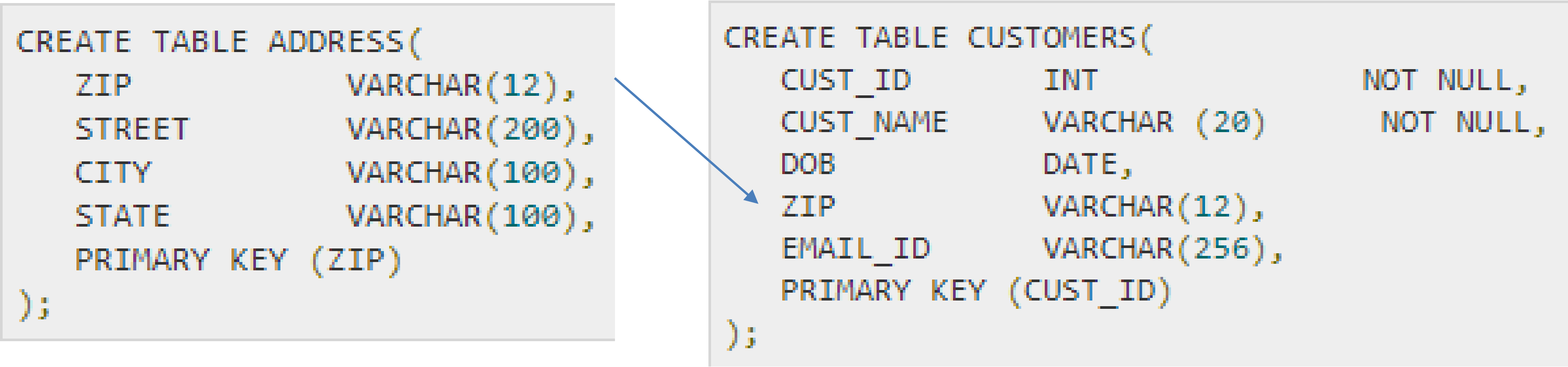
- Database Normalization
  - Third Normal Form (3NF) violation
    - in the following table the street name, city and the state are unbreakably bound to their zip code



```
CREATE TABLE CUSTOMERS(
   CUST_ID       INT                NOT NULL,
   CUST_NAME     VARCHAR (20)        NOT NULL,
   DOB           DATE,
   STREET        VARCHAR(200),
   CITY          VARCHAR(100),
   STATE         VARCHAR(100),
   ZIP           VARCHAR(12),
   EMAIL_ID      VARCHAR(256),
   PRIMARY KEY (CUST_ID)
);
```

RDBMS Concepts

- Database Normalization
  - Third Normal Form (3NF)



```
CREATE TABLE ADDRESS(
   ZIP             VARCHAR(12),
   STREET          VARCHAR(200),
   CITY            VARCHAR(100),
   STATE           VARCHAR(100),
   PRIMARY KEY (ZIP)
);
```

```
CREATE TABLE CUSTOMERS(
   CUST_ID        INT                 NOT NULL,
   CUST_NAME      VARCHAR (20)         NOT NULL,
   DOB            DATE,
   ZIP            VARCHAR(12),
   EMAIL_ID       VARCHAR(256),
   PRIMARY KEY (CUST_ID)
);
```

RDBMS Concepts

- Database Normalization
  - Third Normal Form (3NF)
    - Advantages of removing transitive dependencies
      - the amount of data duplication is reduced and therefore the database becomes smaller.
      - data integrity
    - Data Integrity
      - when duplicated data changes, there is a big risk of updating only some of the data, especially if it is spread out in many different places in the database.
      - For example, if the address and the zip code data were stored in three or four different tables, then any changes in the zip codes would need to ripple out to every record in those three or four tables