**NAME:**

**CWID:**

**Case Study 1:**

1. Using SQL, write Data Definition Language commands to set up the database schema for an Admission Database System for a University with the tables below:
   1. STUDENT(sID, sName, GPA, sizeHS)
   2. COLLEGE(cName, state, enrollment)
   3. Apply(sID, cName, major, decision)

*Hint:*

* + 1. *sID is unique and not null for each Student*
    2. *cName is unique and not null for each College*
    3. *sID, cName and major are unique for each record in the Apply table*
    4. *GPA is a decimal value with at least 2 decimal points*

1. Research (over google) the SQL syntax of the SQL “INSERT INTO” statement and determine the commands to populate the tables with data below.

**COLLEGE**

|  |  |  |
| --- | --- | --- |
| **cName** | **state** | **enrollment** |
| Stanford | CA | 15000 |
| Berkeley | CA | 36000 |
| MIT | MA | 10000 |
| Cornell | NY | 21000 |

**STUDENT**

|  |  |  |  |
| --- | --- | --- | --- |
| **sID** | **sName** | **GPA** | **sizeHS** |
| 123 | Amy | 3.9 | 1000 |
| 234 | Bob | 3.6 | 1500 |
| 345 | Craig | 3.5 | 500 |
| 456 | Doris | 3.9 | 1000 |
| 567 | Edward | 2.9 | 2000 |
| 678 | Fay | 3.8 | 200 |
| 789 | Gary | 3.4 | 800 |
| 987 | Helen | 3.7 | 800 |
| 876 | Irene | 3.9 | 400 |
| 765 | Jay | 2.9 | 1500 |
| 654 | Amy | 3.9 | 1000 |
| 543 | Craig | 3.4 | 2000 |

**Apply**

|  |  |  |  |
| --- | --- | --- | --- |
| **sID** | **cName** | **major** | **decision** |
| 123 | Stanford | CS | Y |
| 123 | Stanford | EE | N |
| 123 | Berkeley | CS | Y |
| 123 | Cornell | EE | Y |
| 234 | Berkeley | Biology | N |
| 345 | MIT | Bioengineering | Y |
| 345 | Cornell | Bioengineering | N |
| 345 | Cornell | CS | Y |
| 345 | Cornell | EE | N |
| 678 | Stanford | History | Y |
| 987 | Stanford | CS | Y |
| 987 | Berkeley | CS | Y |
| 876 | Stanford | CS | N |
| 876 | MIT | Biology | Y |
| 876 | MIT | marine biology | N |
| 765 | Stanford | History | Y |
| 765 | Cornell | History | N |
| 765 | Cornell | Psychology | Y |
| 543 | MIT | CS | N |

**Case Study 2:**

You are provided with the specification requirements below to implement a database management system (DBMS) for “Bank X” to store details about their Branches, Accounts, and Customers.

**SR-1:** A BRANCH has a BranchID that is numeric, a Branch Name, and Branch Address. The BranchID is unique for each branch while Branch Name and Branch Address may not be unique.

**SR-2:** An ACCOUNT has an AccountNo field in numeric characters, Account Type, and Balance. The ACCOUNT\_TYPE is either Savings, Checking, Credit Card, or Mortgage. The AccountNo is unique for each account.

**SR-3:** CUSTOMER has an SSN field that is of numeric characters, along with Name, Phone, Address.

**SR-4:** A Branch has multiple accounts linked to it, while an account can be linked to just one branch.

**SR-5:** A customer can possess no more than 4 accounts in the same branch but of different account types. While accounts are not shared between customers.

**SR-6:** A customer can have multiple accounts in different branches.

**SR-7:** Only the customer’s phone number may be empty (NULL), all fields in the database are mandatory.

1. Using SQL, write **Data Definition Language** (DDL) commands to set up the database schema with the below tables*. Note: Include a primary key for each table*:
   1. BRANCH (*BranchID, BranchName, BranchAddress*)
   2. ACCOUNT (*AccNo, AccTypeID, Branch, SSN, Balance*)
   3. ACCOUNT\_TYPE (*AccTypeID, AccountType*)
   4. CUSTOMER (*SSN, Name, Phone, Address*)
2. Using SQL, write **Data Manipulation Language** (DML) commands to populate the tables with data to illustrate **SR4** to **SR7**.