# Content:

[Requires: 2](#_Toc376817533)

[Exercise 1: ERD 3](#_Toc376817534)

[Exercise 2: Database 3](#_Toc376817535)

[Exercise 3: Data Types 4](#_Toc376817536)

[Exercise 4: Data Types (2) 4](#_Toc376817537)

# Requires:

For the following exercises:

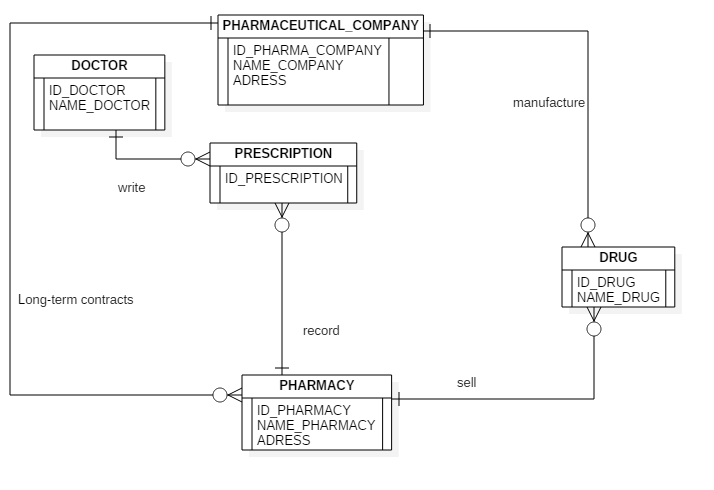
* Print out respectively the screenshots to show the query results.
* SQL scripts for the exercises.
* Pack screenshots and SQL scripts into the zip file named **Assignment1\_AccountName.zip**(for instance: Assignment1\_NamNT.zip)

# Exercise 1: ERD

The following is a description of some data requirements for a chain of pharmacies. Draw the appropriate entity-relationship (E-R) diagram.

* A pharmaceutical company manufactures one or more drugs, and each drug is manufactured and marketed by exactly one pharmaceutical company.
* Drugs are sold in pharmacies. Each pharmacy has a unique identification. Every pharmacy sells one or more drugs, but some pharmacies do not sell every drug.
* Drug sales must be recorded by prescription, which are kept as a record by the pharmacy. A prescription clearly identifies the drug, physician, and patient, as well as the date it is filled.
* Doctors prescribe drugs for patients. A doctor can prescribe one or more drugs for a patient and a patient can get one or more prescriptions, but a prescription is written by only one doctor.
* Pharmaceutical companies may have long-term contracts with pharmacies and a pharmacy can contract with zero, one, or more pharmaceutical companies. Each contract is uniquely identified by a contract number.

Answer:



# Exercise 2: Database

Consider the following relations for a database that keeps track of student enrollment in courses and the books adopted for each course (Primary Key is mark post-fix with #):

STUDENT(SSN#, Name, Major, Bdate)

COURSE(Course#, Cname, Dept)

ENROLL(SSN, Course#, Quarter, Grade)

BOOK\_ADOPTION(Course#, Quarter, Book\_ISBN)

BOOK(Book\_ISBN#, Book\_Title, Publisher, Author)

Draw a relational schema diagram specifying the foreign keys for this schema.

Answer:

|  |  |  |  |
| --- | --- | --- | --- |
| **STUDENT** | | | |
| #SSN | Name | Major | Bdate |

|  |  |  |
| --- | --- | --- |
| **COURSE** | | |
| #Course | Cname | Dept |

|  |  |  |  |
| --- | --- | --- | --- |
| ENROLL | | | |
| #SSN | #Course | Quarter | Grade |

|  |  |
| --- | --- |
| **BOOK\_ADOPTION** | |
| #(Course, Book\_ISBN) | Quarter |

|  |  |  |  |
| --- | --- | --- | --- |
| **BOOK** | | | |
| #Book\_ISBN | Book\_Title | Publisher | Author |

# Exercise 3: Data Types

Below table will have less than 1 million records. It contains information:

|  |  |
| --- | --- |
| Column | Description |
| ID | Primary Key of the table, each time a record is inserted into the table this column is incresed by 1. |
| Name | A string in English |
| Code | An alphanumeric code that has five characters |
| ModifiedDate | The datetime of the last modification. |

You have to choose approxiate data types for these columns in order to minimize the amount of space used to store each row.

Answer:

|  |  |
| --- | --- |
| Column | Data types |
| ID | Int |
| Name | char(30) |
| Code | char(5) |
| ModifiedDate | smalldatetime |

-ID: less than 1 million records => int (2^-31 -> 2^31-1).

-Name: string in english => char(30).

-Code: has five characters => char(5).

-ModifiedDate: smalldatetime.

# Exercise 4: Data Types (2)

Below table will have more than 1 million records. It contains information:

|  |  |
| --- | --- |
| Column | Description |
| ID | Primary Key of the table, each time a record is inserted into the table this column is incresed by 1. |
| Name | A string in any language. |
| BirthDate | Date of birth |
| Gender | Integer, 0: Male, 1: Female, NULL: Unknown |
| IsDeletedFlag | 0 means active, 1 means deleted. |

You have to choose approxiate data types for these columns in order to minimize the amount of space used to store each row.

Answer:

|  |  |
| --- | --- |
| Column | Type |
| ID | int |
| Name | nvarchar(30) |
| BirthDate | smalldatetime |
| Gender | bit |
| IsDeletedFlag | bit |

ID: more than 1 million records => int (-2^31 -> 2^31 – 1)

Name: any language => nvarchar(30)

BirthDate: smalldatetime

Gender: 0: Male, 1: Female, NULL: Unknown => bit

IsDeletedFlag: 0 means active, 1 means deleted => bit