#1

Ans:To know what a composite key is we need to have the knowledge of what a primary key is, a primary key is a column that has a unique and not null value in an SQL table.

Now a composite key is also a primary key, but the difference is that it is made by the combination of more than one column to identify the particular row in the table.

Composite Key:

A composite key is made by the combination of two or more columns in a table that can be used to uniquely identify each row in the table when the columns are combined uniqueness of a row is guaranteed, but when it is taken individually it does not guarantee uniqueness, or it can also be understood as a primary key made by the combination of two or more attributes to uniquely identify every row in a table.

#2

Ans:

Relational database:

If you're creating a project where the data is predictable, in terms of structure, size, and frequency of access, relational databases are still the best choice.

Normalization can help reduce the size of the data on disk by limiting duplicate data and anomalies, decreasing the risk of requiring vertical scaling in future.

Relational databases are also the best choice if relationships between entities are important.

Non-relational databases can store documents within the documents, which helps keep data that will be accessed together in the same place. But if this isn't right for your needs, a relational database is still the answer. For example, if you have a large dataset with complex structure and relationships, embedding might not create clear enough relationships.

The amount of time that RDMSs have been around also means there is wide support available, from tools to integration with data from other systems.

There are many reasons why you'd want to use a non-relational database over a relational database

- Non-relational databases are suitable for both operational and transactional data.
- They are more suitable for unstructured big data.
- Non-relational databases offer higher performance and availability.
- Flexible schema help non-relational databases store more data of varied types that can be changed without major schema changes.

Summary:

In this article, you have learned about relational vs non-relational databases and how they differ from each other. You've also learned the advantages and disadvantages of both types of databases and which database type is most suitable for various projects.

#3

Ans:

Foreign keys

These link two tables together. It is possible to have more than one foreign key in a table, and they can accept a null value. Foreign key values do not need to be unique; duplicate values can be stored in foreign key columns.

Foreign keys do have to link back to columns with unique values. Those columns are frequently primary keys.

What is a foreign key constraint:

Constraints in general are pieces of code that enforce rules and regulations on the data. They create conditions that the data must satisfy.

Foreign key constraints prevent invalid data from being placed into the foreign key column because the data must be one of the values contained in the table where it's directed. A foreign key constraint links a column in one table to a column in another and prevents actions that would destroy the link between two tables.

In the context of the above example, the constraint is a bit of code that ensures a value can only be added to the Orders table if it already exists in the Customers table.

Foreign key problems

Many database users encounter foreign key errors, often because of referential integrity problems. Referential integrity is the accuracy and consistency of data in a relationship. A foreign key might point to data that no longer exists, or the foreign key's data type doesn't match the primary key data type, eroding referential integrity.

Referential integrity can also be corrupted if the foreign key doesn't reference all the data from the primary key. If there's a parent table for Sales which consists of a primary key of the company name, department name and

address, then the child table for Customers must refer to all attributes of the parent table -- not just one or two. If the child table presents a foreign key value that doesn't correspond to a value in the parent table, it corrupts referential integrity. The unmatched value in the child table is sometimes referred to as an orphan record .

In Figure 2, if the CustomerNo column in the Orders table contained the value five, that value would be an orphan because there is no corresponding CustomerNo 5 in the parent table. If a primary key was deleted from the Customers table, referential integrity would be damaged.

Database administrators can define how the database engine handles foreign key problems and lapses in referential integrity using referential integrity constraints. If a user tries to delete or update a key that a foreign key points to, the following user-defined constraints dictate the referential actions the database engine can take in response:

- **`NO ACTION`.** The database engine raises an error and the action is undone.
- **`CASCADE`**. Corresponding rows are updated or deleted in the table with a foreign key when the parent table's corresponding row is deleted or updated.
- **`SET NULL`.** When enabled, SET NULL allows null values in a table originating from ALTER TABLE or CREATE TABLE.

- **`SET DEFAULT`.** The referenced table sets its foreign key value to the default value upon the deletion or updating of the primary table.

#4

Ans:

A database is an organized collection of structured information, or data, typically stored electronically in a computer system. A database is usually controlled by database management system (DBMS) Together, the data and the DBMS, along with the applications that are associated with them, are referred to as a database system, often shortened to just database.

Data within the most common types of databases in operation today is typically modeled in rows and columns in a series of tables to make processing and data querying efficient. The data can then be easily accessed, managed, modified, updated, controlled, and organized. Most databases use structured query language (SQL) for writing and querying data.

MYSQL is an open source relational database management system based on SQL. It was designed and optimized for web applications and can run on any platform. As new and different requirements emerged with the internet, MySQL became the platform of choice for web developers and web-based applications. Because it’s designed to process millions of queries and thousands of transactions, MySQL is a popular choice for ecommerce businesses that need to manage multiple money transfers. On-demand flexibility is the primary feature of MySQL.

MySQL is the DBMS behind some of the top websites and web-based applications in the world, including Airbnb, Uber, LinkedIn, Facebook, Twitter, and YouTube.

#5

Ans:

CREATE table student(

a. Name:Varchar(30),
b. Roll:int,
c. Class:Varchar(20),
d. Blood group: varchar(20),
e. Contact No:varchar(15),
f. Result:varchar(20),
g. Date of Birth:DATE,
h. Age:int,

);

#6

Ans:

CREATE table student(

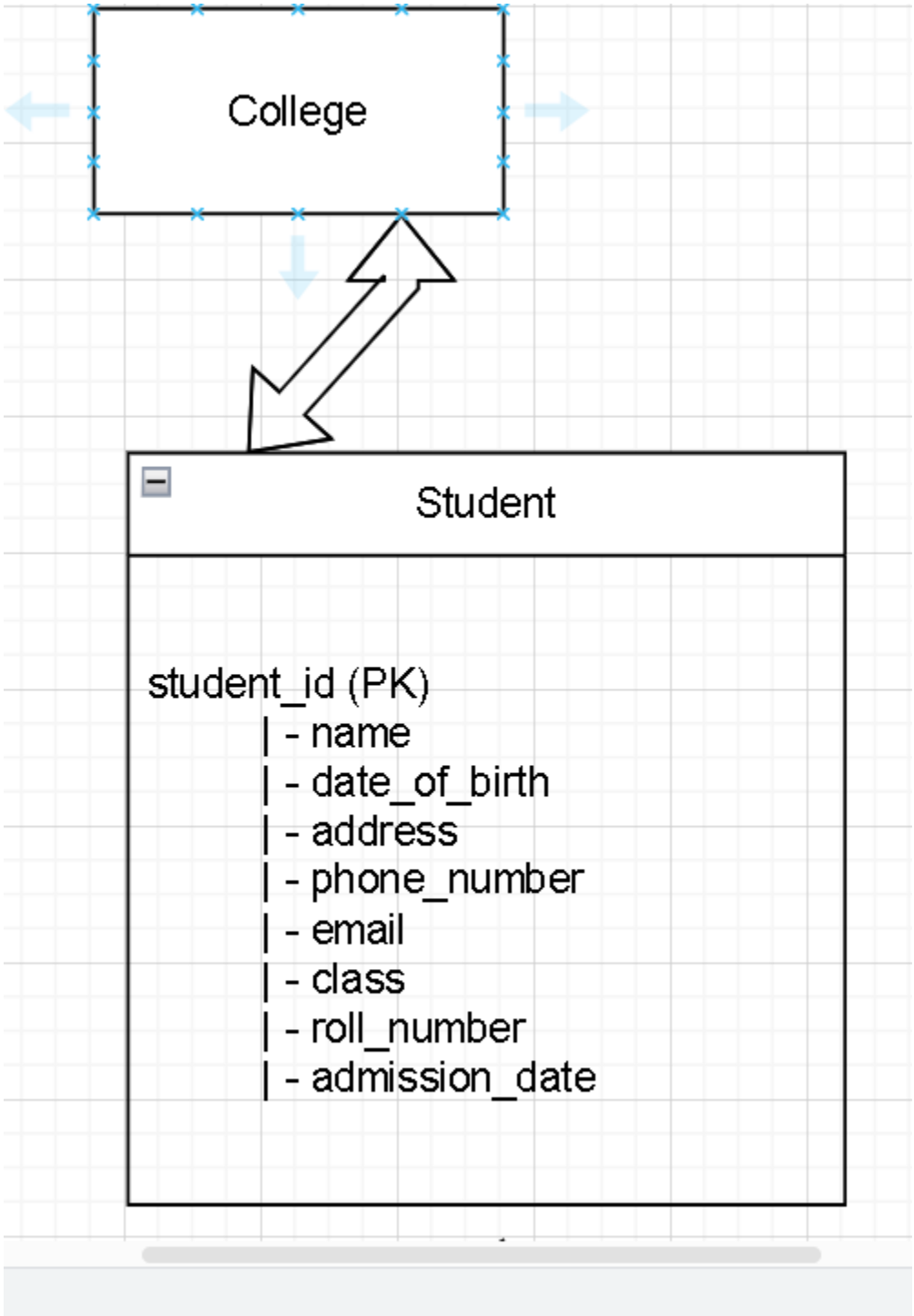
id INT AUTO_INCREMENT PRIMARY KEY,

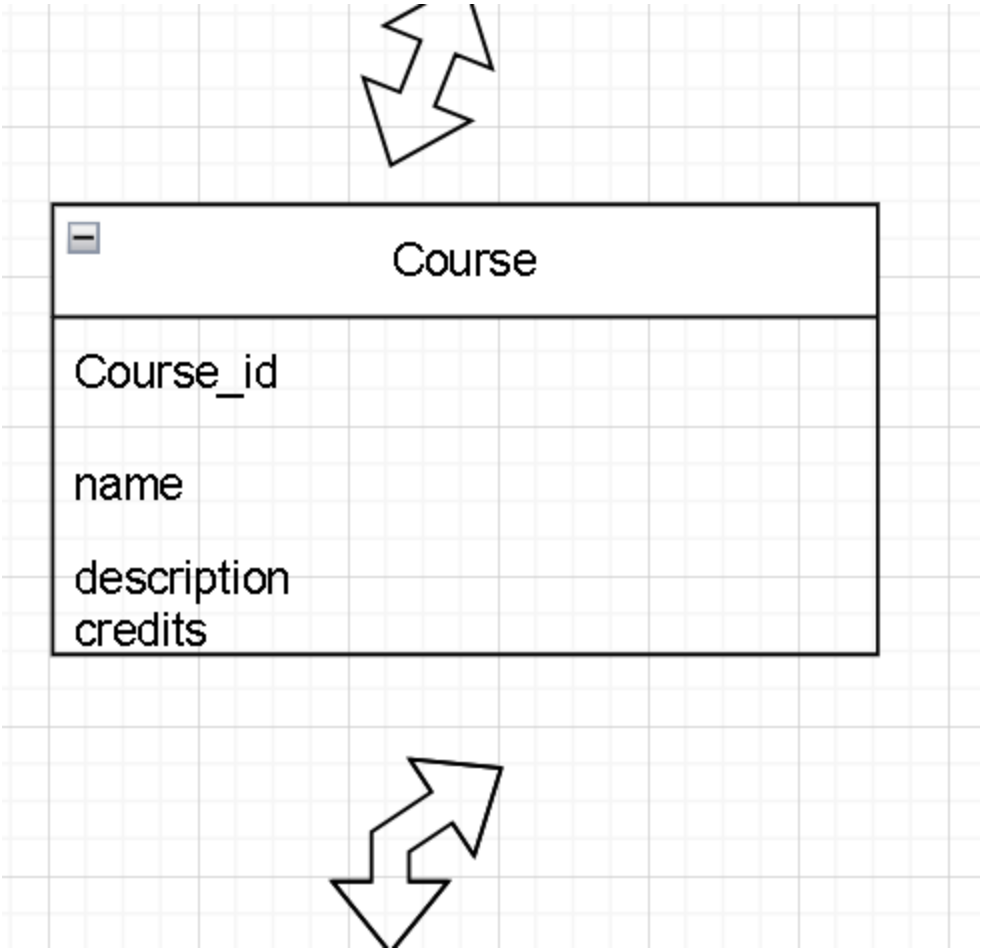
Name:Varchar(30),
Roll int,
Class Varchar(20),
Blood group varchar(20),
Contact No varchar(15),
Result varchar(20),
Date of Birth:DATE,
Age int,

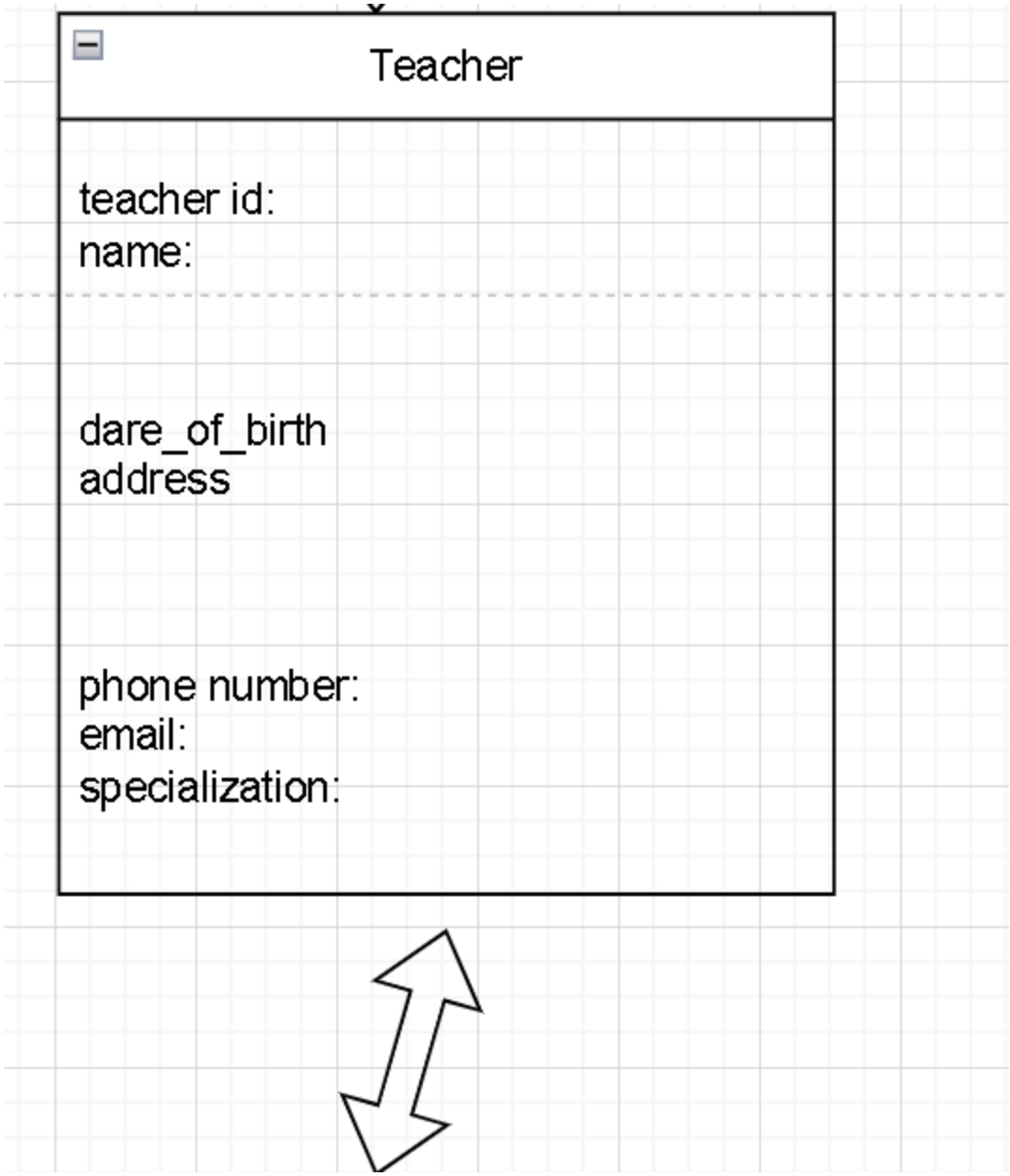
);

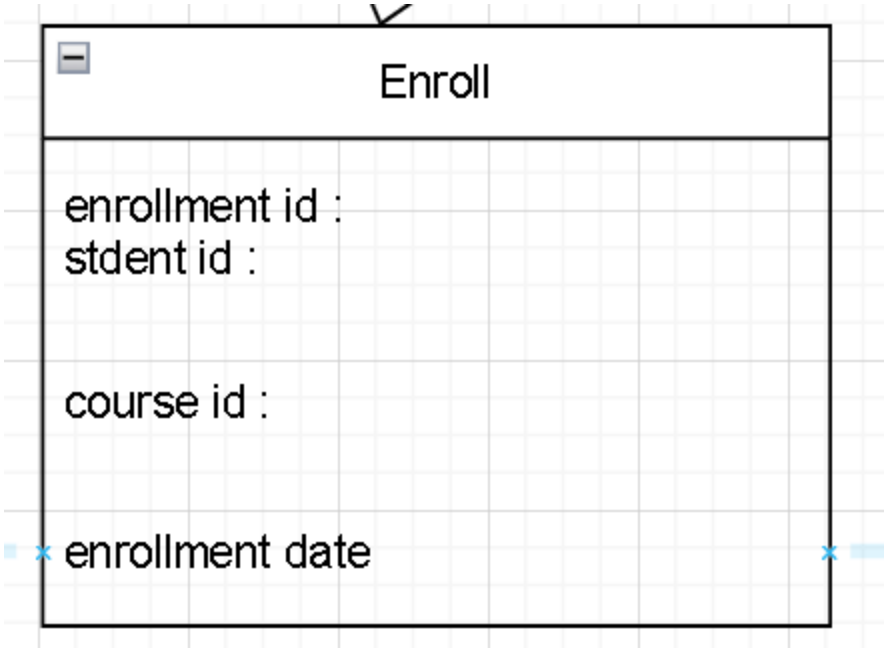
#7

Ans:


College
Student
student_id (PK)
| - name
| - date_of_birth
| - address
| - phone_number
| - email
| - class
| - roll_number
| - admission_date


Course
Course_id
name
description
credits


Teacher
teacher id:
name:
dare_of_birth
address
phone number:
email:
specialization:



#8
Ans:

Ans:

```
CREATE table student(

id INT AUTO_INCREMENT PRIMARY KEY,

                Name Varchar(30),
                Roll int,
                Class Varchar(20),
                Blood group  varchar(20),
                Contact No varchar(15),
                Result varchar(20),
                Date of Birth DATE,
                Age int,
);

-- Rename the table from "student" to "new_student_table"

ALTER TABLE student RENAME TO new_student_table;
```

```
-- Delete the table "new_student_table"

DROP TABLE new_student_table;
```