

# CS301 – Database Management System

20bcs043 – Dipesh Mishra

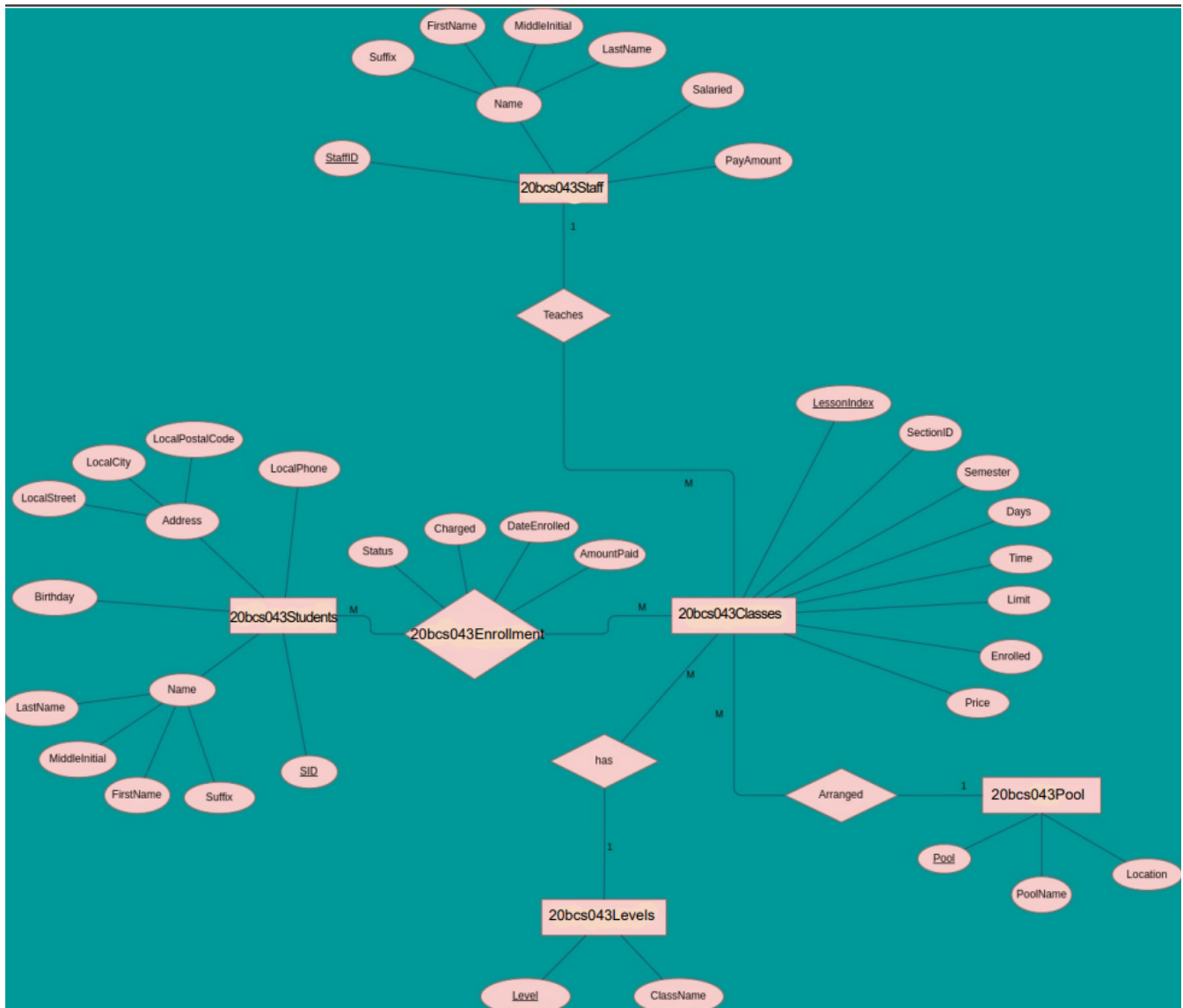
## Questions :

**1. Write the schema for all the relations and design a Conceptual Data model (ERD)- Use online Tool or any other DBMS software . Submit a screenshot of the Conceptual Data model with all the descriptions as a PDF on the GitHub (Note: While creating the data models, you are instructed to replace the letter tbl with your roll number in all the Tables)**

### => Schema :-

- 20bcs043Levels (Level, ClassName)
- 20bcs043Pool (Pool, PoolName, Location)
- 20bcs043Staff (StaffID, FirstName, MiddleInitial, LastName, Suffix, Salaried, PayAmount)
- 20bcs043Classes (LessonIndex, Level, Pool, Instructor, SectionID, Semester, Days, Time, Limit, Enrolled, Price)
- 20bcs043Enrollment (LessonIndex, SID, Status, AmountPaid, DateEnrolled)
- 20bcs043Students (SID, FirstName, MiddleInitial, LastName, Suffix, Birthday, LocalStreet, LocalCity, LocalPostCode, LocalPhone)

### => Conceptual Data Model :-



## 2. Identify degree and cardinality in the conceptual data model.

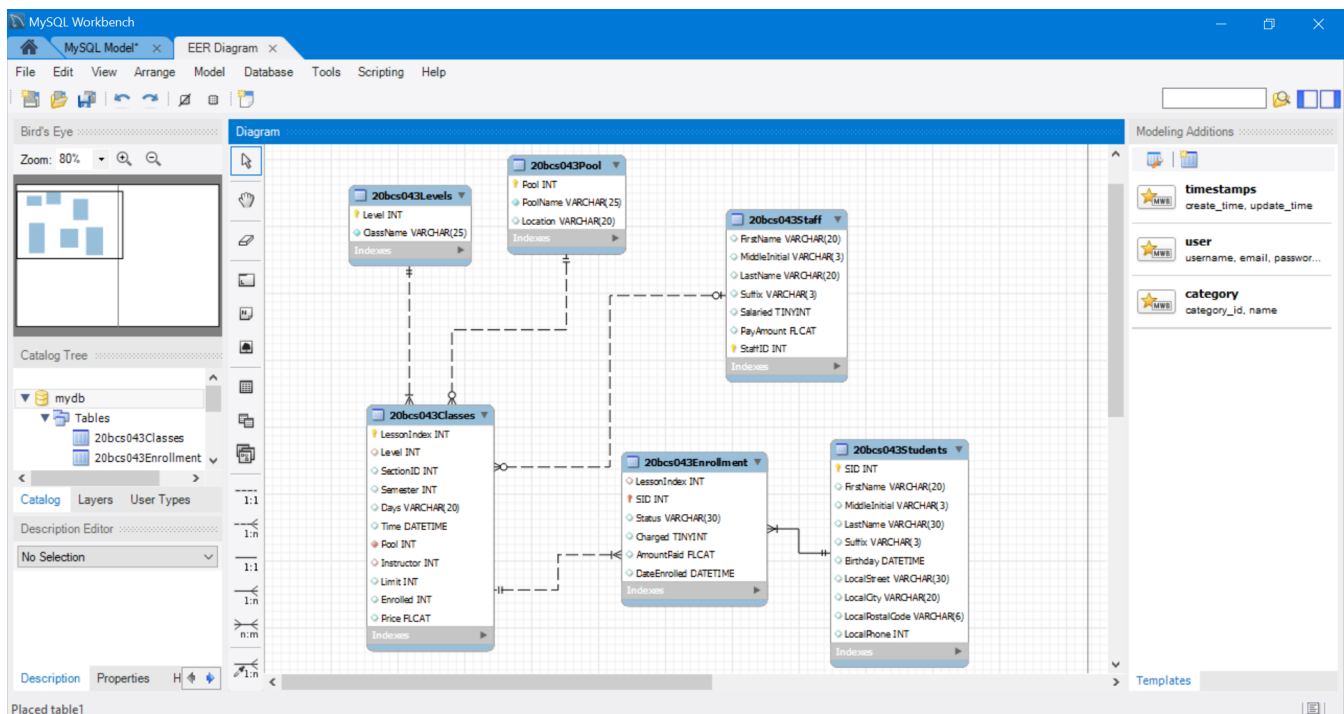
- 20bcs043Staff [Optional 1-Optional Many] 20bcs043Classes
- 20bcs043Pool [Mandatory 1-Optional Many] 20bcs043Classes
- 20bcs043Levels[Mandatory 1-Mandatory Many] 20bcs043Classes
- 20bcs043Classes [Mandatory 1-Mandatory Many] 20bcs043Enrollment
- 20bcs043Enrollment [Mandatory Many- Mandatory 1] 20bcs043Students

## 3.Assume the following rules and draw Physical Data model (ERD). Submit a screenshot of the Physical Data model with all the descriptions as a PDF on the GitHub.

*a. A pool may or may not ever have a class.*

- b. The levels table must always be associated with at least one class.*
- c. The staff table may not have ever taught a class.*
- d. All students must be enrolled in at least one class.*
- e. The class must always be associated with an existing level.*
- f. The class must have a valid pool.*
- g. The class may not have an instructor assigned.*
- h. The class must always be associated with an existing level.*

=> **Physical Data Model :-**



#### ***4. List the weak entity, if exists? Convert them to Strong entity wherever possible in your Physical data model. Create additional Tables if required.***

In the above Diagram 20bcs043Enrollment is an associative entity since it's a M:N relation between 20bcs043Classes and 20bcs043Students, with added attributes. Associative Entities can also be called as weak entities. The table "Enrollment" has no meaning without Classes and Students since you cannot "enroll" without either of the objects. It needs an instance of the Parent Entities to create an instance of the enrolment Table. The enrolment table consists of a composite key which is made from the primary key of Classes, Students.

In this case we cannot make it a strong entity by just adding a primary key to "20bcs043Enrolment" as it will be redundant as we can uniquely identify every row by using the composite key.

***5. Physical data model should have minimal scope for data redundancy. Highlight such data redundancy scenarios in your Physical data model Table if it exists.***

There is no data redundancy present in the table since there are no two instances of the same person in two different tables representing the same data. The database however can be improvised using generalization. A people entity can be added to include "FirstName", "MiddleInitial", "LastName" and add a relation to both Instructor and Student