ENSF 300 Final Project: Arts Museum

Group Name: Nori Elite team-50
Group Number: 50

Link: https://github.com/Maan-Khedr-ENSF-300/museum-project-team-50

Members:
Eric Mei | UCID: 30142893
Lionel Hasan | UCID: 30137476
Liam Mah | UCID: 30144671
Theodore Hoang | UCID: 30141708

Task Distribution Table	
Person	**Tasks Completed**
Eric	- Fill database with data - EERD to Relational Diagrams - Create roles and users (guest, data_entry, admin) - Queries: (6), (7) - Function: command (admin function for executing an SQL command directly) - Function: delete (data entry user function for deleting tuples)
Lionel	- EERD Diagram + Design Assumptions - Create tables for SQL file - Queries: (1) - Function: new_script (admin function for executing an SQL script) - Function: add (data entry function for adding/inserting tuples)
Theodore	- EERD to Relational Diagrams - Create tables for SQL file - Foreign key constraints (with cascade) - Queries: (4) - Function: search (guest function for interfacing and querying data)
Liam	- EERD Diagram + Design Assumptions - Create tables for SQL file - Queries: (2), (3), (5) - Mysql connection for different roles - General layout for interfacing file (menus into specialized functions) - Function: update (data entry function for updating attributes within tuples)

Note: Every group member is responsible for double checking the entire project.

Additional Installations: None (just regular mysql.connector)

Additional Software Notes: The "ARTMUSEUM.sql" script must be ran before the main Python interface is run to avoid errors.

Login Information:

Admin:
User: db_admin
Password: adminpass

Data Entry User:
User: data_entry
Password: dataentrypass

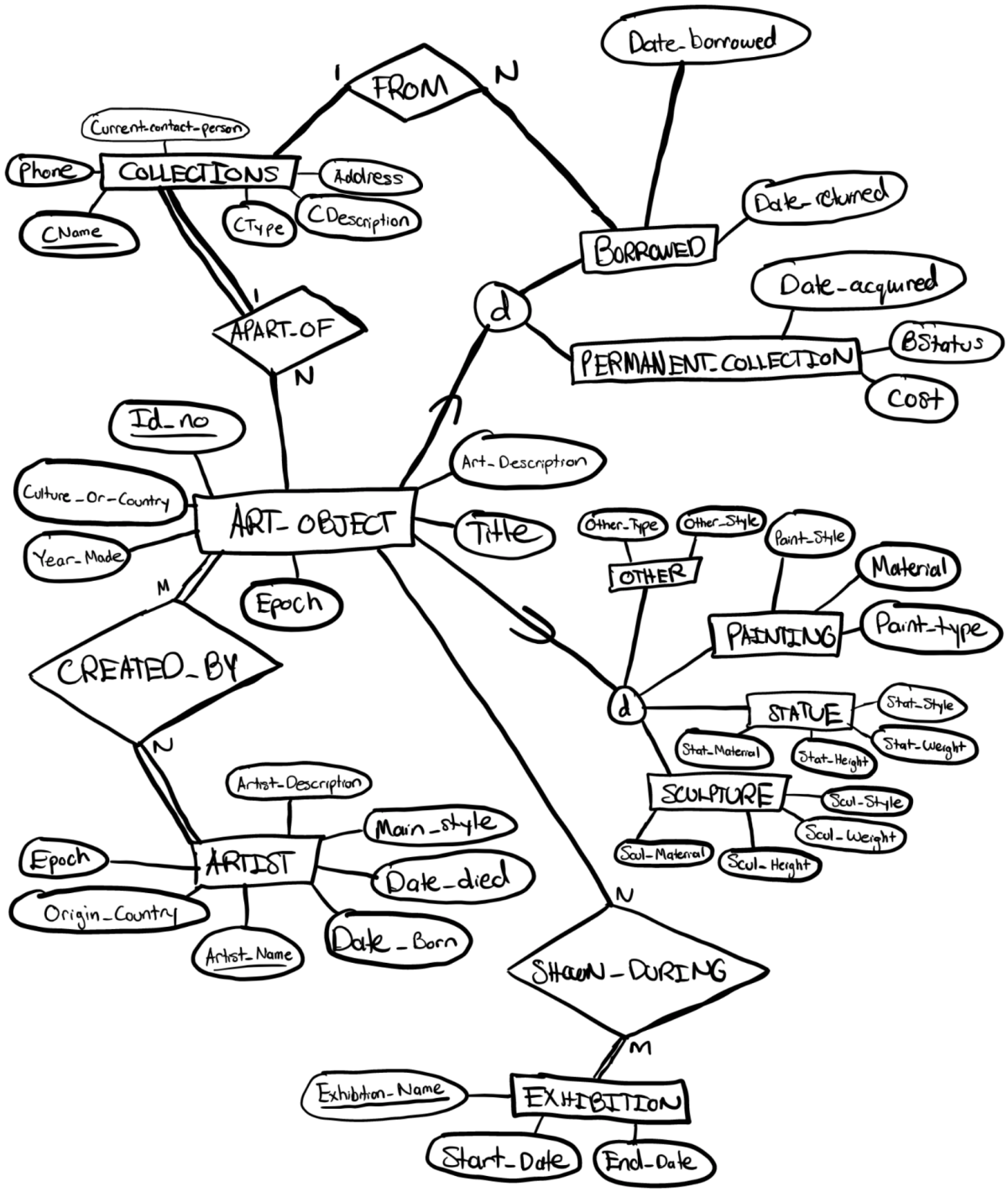
Guest:
User: guest
No Password is required

Note: *Regular username "root" and localhost password can be used to sign in for admin and data entry users.*

Extra Requirements/Handling:

- All bonus point questions were completed
- Additional error and exception handling was added
- Foreign key constraints for every foreign key to maintain referential integrity (set to cascade)

EERD Diagram:



Design Decisions and Assumptions:

In this design we decided to keep names as just a single name, rather than a composite attribute that could be broken down into a first name and last name. This was done in order to keep the database more simple, as the name would be used as a key in some cases.

In the CREATED_BY relationship, we assumed that many artists could work on many art objects. This is because artists can collaborate on art projects, and they are not limited to producing just one art object (e.g. they could produce many art objects contributing to a collection). Each artist must have produced at least one art object, and each art object must have an artist. As a result of this we decided on total participation on both ends of the relationship.

In the APART_OF relationship, we decided that many art objects can contribute to a single collection. This is because in the real world, collections are often used as a form of categorization of things. Logically, this makes the most sense. It was decided that not every art object has to belong to a collection (it can exist as an individual object). Each collection must consist of art objects since it does not make sense to have a collection of nothing. As a result, there is total participation on the end of COLLECTIONS, but partial participation on the ends of ART_OBJECT.

In the SHOWN_DURING relationship, we decided on a many to many relationship because many art objects can be shown in a multitude of exhibitions. Each exhibition must display objects; however, not all art objects have to be shown during each exhibition. Therefore there is total participation on the end of the exhibition, but partial participation on the end of the art object.

When categorizing art objects into BORROWED and PERMANENT_COLLECTION, we decided on a disjoint connection. This is because an ART_OBJECT cannot be both borrowed, and owned (part of the permanent collection) at the same time. It can only be borrowed or owned.

When categorizing art objects into painting, statue, sculpture, or other, we decided on a disjoint connection. This is because a single art object cannot be many things at the same time. For example, it does not make sense to have a painting and statue at the same time in a single art object.

Lastly, when creating the FROM relation between collections, and borrowed. Borrowed art objects do not have to belong in a collection, and a collection can consist of objects that are not borrowed. As a result, we decided on partial participation on both sides. Many borrowed objects can belong in a single collection, so we decided on a 1:N relationship from COLLECTIONS to BORROWED.

Relational Diagram: