# C2- S7 - PRACTICE

*NOTE: check your* ***THEORY slides*** *to answer those questions!*

# EXERCISE 1 – THE COMPANY DATABASE

# A company database needs to store information about:

* **Employees** that are described by their **salary** and **phone** and **email address** and **sex** and **job**
* **Departments** that are described by their **name** and their **budget**
* **Children** of employees that are described by their **name** and **age**
* **Project** of each department that are described by their **name**, **budget** and **deadline**

Here is some more information on how works the company:

* Employees work in departments. One employee can work for different departments.
* One department can have many employees working in it.
* Each child has only one parent that works in the company.
  + We are not interested in information about a child once the parent leaves the company.
* One employee can work on many project
* Many employees can work on one project
* A project is assigned to one department
* One department can have assigned many different projects

**Q1 –** Design the ERD Physical Model of the company database

|  |  |  |
| --- | --- | --- |
| Department | | |
| PK | DepartmentID  Name  Budget | Int  Varchar (100)  Int |

|  |  |  |
| --- | --- | --- |
| Children | | |
|  | ChildrenID  Name  Age  EmployeeID  many | Int  Varchar (100)  Int |

one

one

|  |  |  |
| --- | --- | --- |
| Employee-Department  many | | |
| CK | EmployeeID  DepartmentID | Int  Int |

many

|  |  |  |
| --- | --- | --- |
| Project  many | | |
| PK  FK | ProjectID  Name  Budget  Deadline  DepartmentID | Int  Varchar (100)  Int  Datetime  Int |

|  |  |  |
| --- | --- | --- |
| Employee  one | | |
| PK | EmployeeID  Salary  Phone  Address  Sex  job | Int  Int  Int  Varchar (200)  Char (1)  Varchar(100) |

one

one

|  |  |  |
| --- | --- | --- |
| Employee-Project | | |
| CK  many | EmployeeID  ProjectID | Int  Int |

one

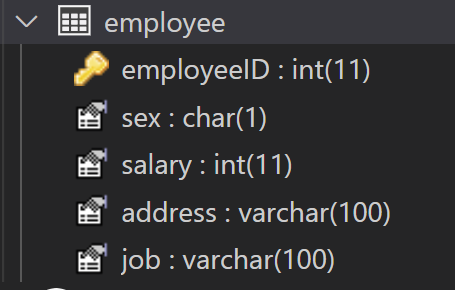
many

**Q2 –** Implement this database in MySQL

​​​​​​​​​​​​​​​​

**Employee table**

CREATE TABLE IF NOT EXISTS employee (

    employeeID int NOT NULL AUTO\_INCREMENT,

    salary int,

    sex char(1) NOT NULL,

    address varchar(255),

    phone int,

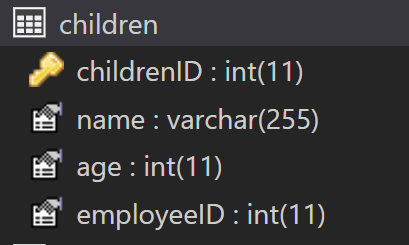
    job varchar(100),

    PRIMARY KEY (employeeID)

);

**Children table**

CREATE TABLE IF NOT EXISTS children (

    childrenID INT PRIMARY KEY AUTO\_INCREMENT,

    name VARCHAR(255) NOT NULL,

    age INT NOT NULL,

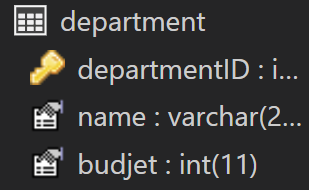
    employeeID int,

    FOREIGN KEY (employeeID)

        REFERENCES employee (employeeID)

        ON DELETE CASCADE

);

**Department table**

CREATE TABLE IF NOT EXISTS department (

    departmentID INT PRIMARY KEY AUTO\_INCREMENT,

    name VARCHAR(255) NOT NULL,

    budjet INT NOT NULL

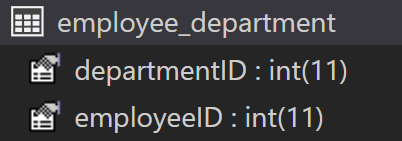
);

**Employee\_Department**

CREATE TABLE IF NOT EXISTS employee\_department (

    departmentID INT ,

    employeeID int,

    FOREIGN KEY(departmentID)

        REFERENCES department(departmentID),

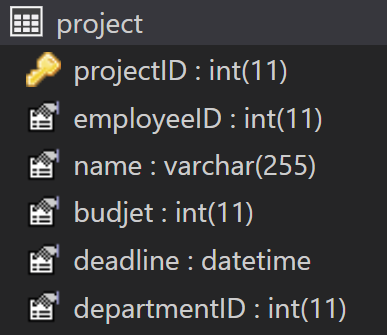
    FOREIGN KEY(employeeID)

        REFERENCES employee(employeeID)

);

**Project table**

CREATE TABLE IF NOT EXISTS project (

    projectID INT AUTO\_INCREMENT PRIMARY KEY ,

    employeeID int,

    name varchar (255) NOT NULL,

    budjet int not NULL,

    deadline datetime NOT NULL,

    departmentID int,

    FOREIGN KEY (departmentID)

        REFERENCES department(departmentID)

);

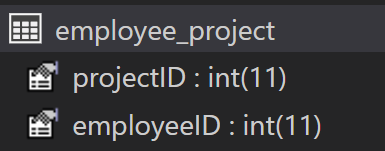
Employee-project

CREATE TABLE IF NOT EXISTS employee\_project (

    projectID INT ,

    employeeID int,

    FOREIGN KEY(projectID)

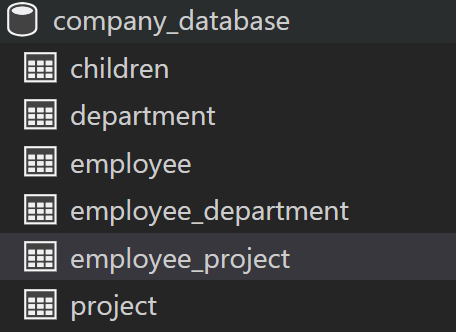
        REFERENCES project(projectID),

    FOREIGN KEY(employeeID)

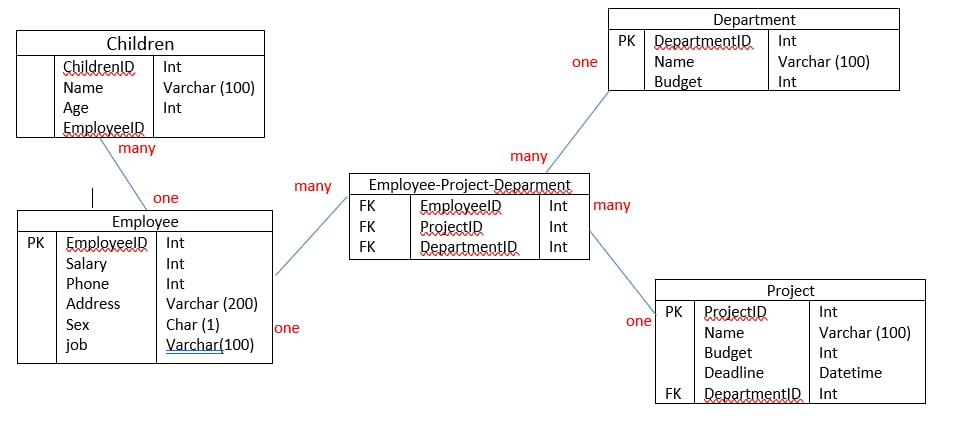
        REFERENCES employee(employeeID)

);

**Company database**



**The best ERD**



# EXERCISE 2 – THE ARTBASE DATABASE

An application named ARTBASE want to sell a product for art galleries. It is an application that stores in a database all the information that an art gallery needs to keep, to work effectively.

* Gallery keep information about artists, their names (which are unique), birthplaces, age, and style of art.
* For each piece of artwork, the artist, the year it was made, its unique title, its type of art (e.g., painting, lithograph, sculpture, photograph), and its price must be stored. Pieces of artwork are also classified into groups of various kinds, for example, portraits, works by Picasso, or works of the 19th century.
* a given piece may belong to more than one group.
* Each group is identified by a name (like those just given) that describes the group.
* Finally, galleries keep information about customers. For each customer, galleries keep that person’s unique name, address, total amount of dollars spent in the gallery (very important!), and the artists and groups of art that the customer tends to like.

**Q1 –** Design the ERD Physical Model of the company database

|  |  |  |
| --- | --- | --- |
| ArtWork\_Group | | |
| PK | art\_work\_id  art\_group\_id | INT  Int |

|  |  |  |
| --- | --- | --- |
| ArtType | | |
| PK | art\_type\_id  name | INTINT  VARCHAR(200) |

one

one

many

many

|  |  |  |
| --- | --- | --- |
| Artwork  many  one | | |
| PK  FK  many  FK  FK  FK | art\_work\_id  title  year  artist\_id  art\_type\_id | INT  VARCHAR(200)  DATE  INT  INT |

|  |  |  |
| --- | --- | --- |
| Art-group  one | | |
| PK | art\_group\_id  name | INT  VARCHAR(200) |

|  |  |  |
| --- | --- | --- |
| Artist  many | | |
| PK  FK  FK | artist\_id  name  birth\_place  age  art\_type\_id | INT  VARCHAR(200)  VARCHAR(200)  INT  INT |

one

one

one

|  |  |  |
| --- | --- | --- |
| Customer\_art\_group  many | | |
| FK  FK | Customer\_id  Art-group\_id  many | Int  Int |

|  |  |  |
| --- | --- | --- |
| Customer | | |
| PK  one | Customer\_id  Name  Address  Amount | INT  VARCHAR(200)  VARCHAR(200)  FLOAT |

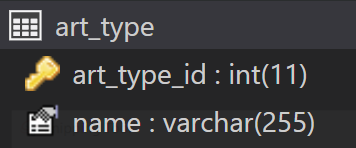
|  |  |  |
| --- | --- | --- |
| Customer\_Artist  many | | |
| FK  FK | Customer\_id  Artist\_id | Int  Int |

many

one

**Q2 –** Implement this database in MySQL

**Art\_type table**

CREATE TABLE IF NOT EXISTS art\_type(

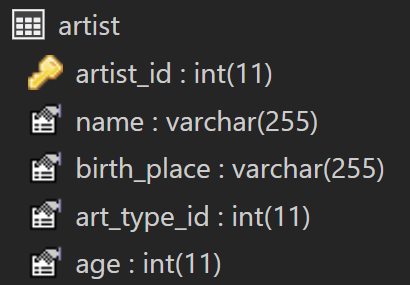
    art\_type\_id int AUTO\_INCREMENT PRIMARY KEY,

    name varchar (255) NOT NULL

);

**Artist table**

CREATE TABLE IF NOT EXISTS artist(

    artist\_id int AUTO\_INCREMENT PRIMARY KEY,

    name varchar (255) NOT NULL,

    birth\_place varchar(255),

    art\_type\_id int,

age int,

    FOREIGN KEY(art\_type\_id)

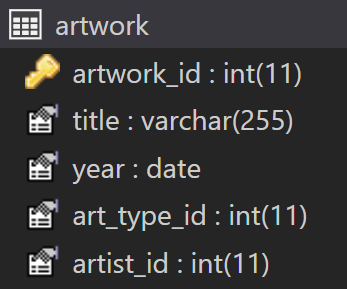
REFERENCES art\_type(art\_type\_id)

);

**Artwork table**

CREATE TABLE artwork (

    artwork\_id int NOT NULL PRIMARY KEY,

    title varchar(255) UNIQUE,

    year date UNIQUE,

    art\_type\_id int,

    artist\_id int,

    FOREIGN KEY(art\_type\_id)

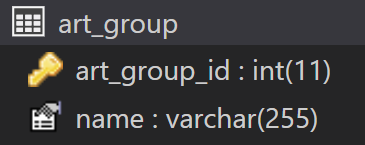
        REFERENCES art\_type (art\_type\_id),

    FOREIGN KEY(artist\_id)

        REFERENCES artist(artist\_id)

);

**Art\_group table**

CREATE TABLE IF NOT EXISTS art\_group (

    art\_group\_id int AUTO\_INCREMENT PRIMARY KEY,

    name varchar (255)  not null

);

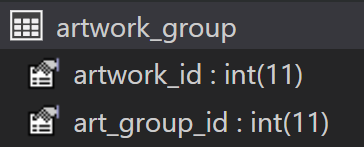
**Artwork\_group table**

CREATE TABLE artwork\_group (

    artwork\_id int ,

    art\_group\_id int,

    FOREIGN KEY(artwork\_id)

        REFERENCES artwork (artwork\_id),

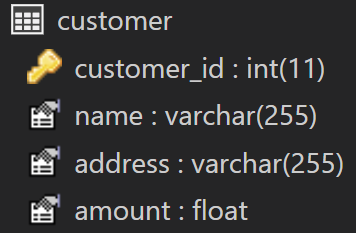
    FOREIGN KEY(art\_group\_id)

        REFERENCES art\_group(art\_group\_id)

);

**Customer table**

CREATE TABLE customer (

    customer\_id int AUTO\_INCREMENT PRIMARY KEY ,

    name varchar (255) not null,

    address varchar (255),

    amount float not null

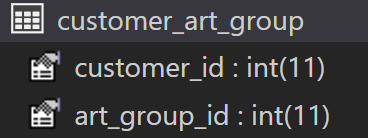
);

**Customer art group table**

CREATE TABLE customer\_art\_group (

    customer\_id int ,

    art\_group\_id int,

    FOREIGN KEY(customer\_id)

        REFERENCES customer (customer\_id),

    FOREIGN KEY(art\_group\_id)

        REFERENCES art\_group(art\_group\_id)

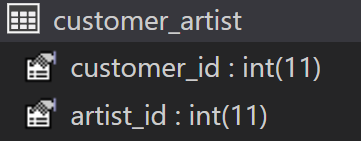
);

**Customer artist table**

CREATE TABLE customer\_artist (

    customer\_id int ,

    artist\_id int,

    FOREIGN KEY(customer\_id)

        REFERENCES customer (customer\_id),

    FOREIGN KEY(artist\_id)

        REFERENCES artist(artist\_id)

);