SUBMITTED BY :CS-5A SUBMITTED TO:Ms.SHIVANI

1.ANUSHREE GHOSH(2100290120041)

2.AKANKSHA GUPTA(2100290120022)

3.ANANYA SRIVASTAVA(2100290120028)

4.APURWA KUMARI(2100290120043)

5.ANNU(2100290120032)

6.AYUSHI CHAUHAN(2100290120063



**ART GALLERY**

**DATABASE MANAGEMENT**

**ART GALLERY**

**DATABASE MANAGEMENT**

**The main objective of creating an Art Gallery database project is**

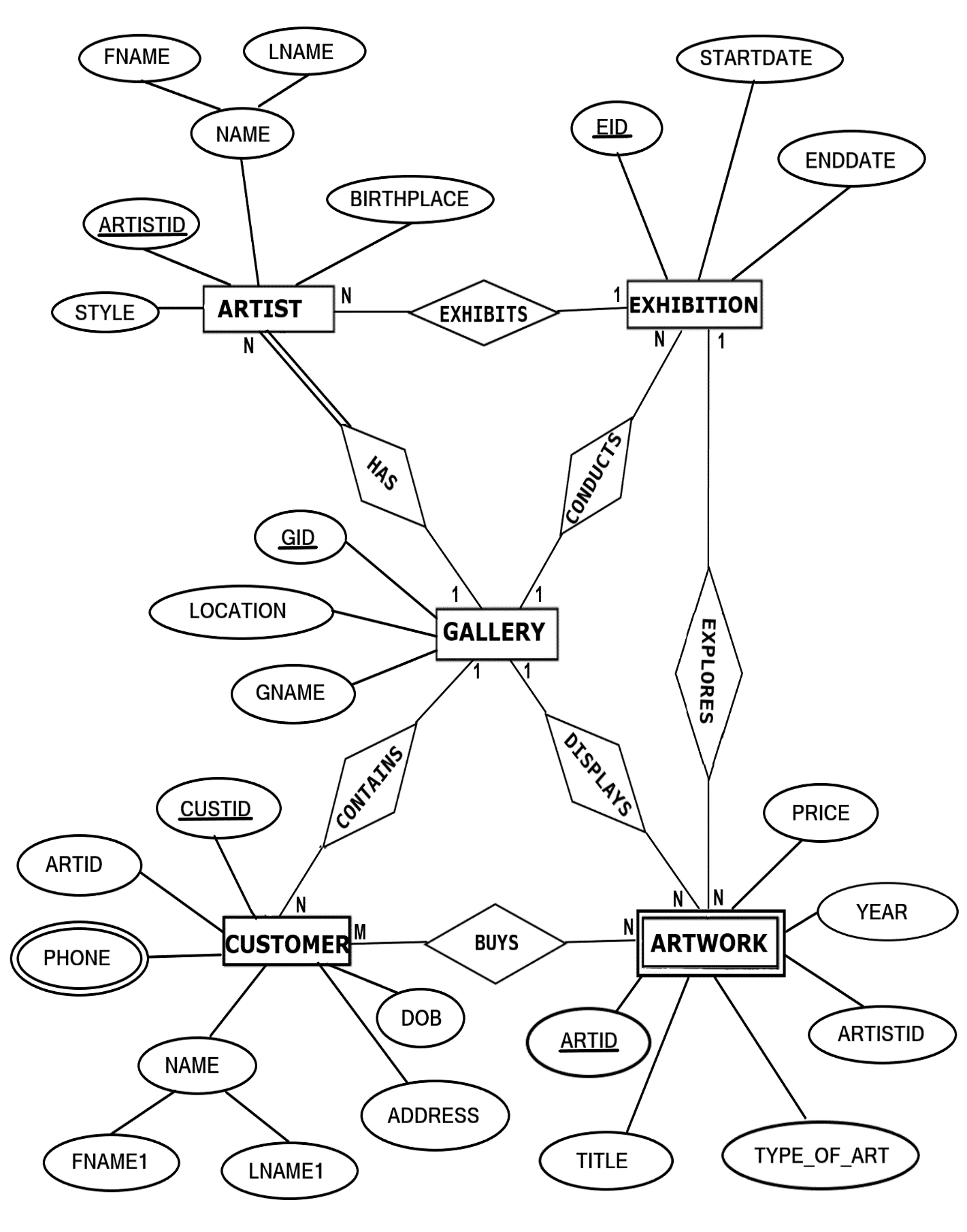
* To manage the details of the gallery, exhibition, artwork, and artist. The purpose of the project is to build an application program to reduce manual work.
* To tracks all the details about the sales of the artwork, the customer that bought it, etc. It manages the information about the artwork. Provides information and description of the artworks left, thereby increasing the efficiency of managing the gallery. The organization can maintain a computerized record of the artwork present in the gallery.
* To helps in the utilization of the resources in an effective manner. It maintains a list of all the customers and the various artwork that they have bought and the money that have invested in each.
* To maintains the record of exhibitions and various sales made during it. The objective of developing such a computerized system is to reduce the paperwork and safe of time in art gallery database management, thereby increasing the efficiency and decreasing the work load.

Developing such a computerized system reduces the paperwork and save of time in art gallery database management, thereby increasing efficiency and decreasing the workload.

**IMPLEMENTATION**

**4.1 ER DIAGRAM**

1. An **entity-relationship model (ER Model)** describes inter-related things of interest in a specific domain of knowledge. An ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between instances of those entity types.
2. An entity may be defined as a thing capable of an independent existence that can be uniquely identified. An entity is an abstraction from the complexities of a domain.
3. Attributes are drawn as ovals and are connected with a line to exactly one entity or relationship set.
4. An entity relationship model, also called an entity-relationship (ER) diagram, is a graphical representation of entities and their relationships to each other, typically used in computing in regard to the organization of data within databases or information systems.
5. Cardinality constraints are expressed as follows:
6. A double line indicates a participation constraint, totality or subjectivity: all entities in the entity set must participate in at least one relationship in the relationship set.
7. An arrow from entity set to relationship set indicates a key constraint, i.e. injectivity: each entity of the entity set can participate in at most one relationship in the relationship set.
8. A thick line indicates both, i.e. bijectivity: each entity in the entity set is involved in exactly one relationship.
9. An underlined name of an attribute indicates that it is a key: two different entities or relationships with this attribute always have different values for this attribute.

**FIGURE 1.1: ER DIAGRAM of ART GALLERY DATABASE**

**1.2 MAPPING OF ER DIAGRAM TO RELATIONS**

**STEP 1: Mapping of Regular Entities**

For each regular entity type E in the ER schema, create relation R that includes all simple attributes of E.

**GALLERY**

|  |  |  |
| --- | --- | --- |
| **GID** | GNAME | LOCATION |

**EXHIBITION**

|  |  |  |
| --- | --- | --- |
| **EID** | STARTDATE | ENDDATE |

**ARTIST1**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ARTISTID** | FNAME | LNAME | BIRTHPLACE | STYLE |

**CUSTOMER1**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CUSTID** | **ARTID** | FNAME1 | LNAME1 | ADDRESS | PHONE | DOB |

**FK (FOREIGN KEY)**

**STEP 2 : Mapping of Weak Entity Types**

**ARTWORK**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ARTID** | **ARTISTID** | TITLE | TYPE\_OF\_ART | YEAR | PRICE |

**FK**

**STEP 3: Mapping of 1:1 Relationship**

Identify the relation S that represents the participating entity type at the 1-side of the relationship type.

Include as foreign key in S the primary key of the relations T that represents the other entity type participating in R.

For each binary 1:1 relationship type R in ER schema, identify the relations S and T that correspond to the entity types participating in R if any.

There are **no** 1:1 relationship.

**STEP 4 : Mapping of 1:N Relationship**

**EXHIBITION**

|  |  |  |  |
| --- | --- | --- | --- |
| **EID** | STARTDATE | ENDDATE | **GID** |

**FK**

**ARTIST**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ARTISTID** | FNAME | LNAME | BIRTHPLACE | STYLE | **EID** | **GID** | **CUSTID** |

**FK FK FK**

**CUSTOMER**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CUSTID** | **ARTID** | FNAME1 | LNAME1 | ADDRESS | DOB | **GID** |

**FK FK**

**ARTWORK**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ARTID** | **ARTISTID** | TITLE | TYPE\_OF\_ART | YEAR | PRICE | **EID** | **GID** |

**FK FK FK**

**STEP 5 : Mapping of M:N Relationship**

Create a new relation S to represent R.

Include as foreign key attributes in S the primary key of the relations that represents the participating entity types their combination will form the primary key of S.

Also, include any simple attributes of the M:N relationship type as attributes of S.

**STEP 6: Mapping of Multi-Valued Attributes**

For each multivalued attributes A, create a new relation R. This relation R will include an attribute corresponding to A, plus the primary key attribute K-as a foreign key in R-of the relation that represents the entity type of relationship type that has A as an attribute.

The Primary Key of R is the combination of A and K. If the multivalued attribute is composite, we include its simple components.

**1.3 SCHEMA DIAGRAM**

**GALLERY**

|  |  |  |
| --- | --- | --- |
| **GID** | GNAME | LOCATION |

**EXHIBITION**

|  |  |  |  |
| --- | --- | --- | --- |
| **EID** | STARTDATE | ENDDATE | **GID** |

**CUSTOMER**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CUSTID** | **ARTID** | FNAME1 | LNAME1 | ADDRESS | DOB | **GID** |

**ARTIST**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ARTISTID** | FNAME | LNAME | BIRTHPLACE | STYLE | **EID** | **GID** | **CUSTID** |

**ARTWORK**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ARTID** | **ARTISTID** | TITLE | TYPE\_OF\_ART | YEAR | PRICE | **EID** | **GID** |

**CONTACTS**

|  |  |
| --- | --- |
| **CUSTID** | PHONE |

**FIGURE 1.3: SCHEMA DIAGRAM**

**1.4 NORMALIZE THE RELATIONS**

Database normalization, or simply normalization, is the process of organizing the columns(attributes) and tables(relations) of a relational database to reduce data redundancy and improve data integrity. Normalization involves arranging attributes in relations based on dependencies between attributes.

1. **First Normal Form**

As per First normal form, no two rows of data must contain repeating group of information. Each set of columns must have a unique value, such that multiple columns cannot be used to fetch the same row. Each table should be organized into rows, and each row should have a primary key that will distinguishes it as unique.

**Example:**

**GALLERY**

|  |  |  |
| --- | --- | --- |
| **GID** | GNAME | LOCATION |

All the tables in the database are normalized to 1NF as all the attributes are atomic.

1. **Second Normal Form (2NF)**

A table is in 2NF if it is in 1NF and if all non-key attributes are fully functionally dependent on all of the key.

**Example:**

**CUSTOMER**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CUSTID** | **ARTID** | FNAME1 | LNAME1 | ADDRESS | DOB | **GID** |

**FD1**

**FD1**

|  |  |  |  |
| --- | --- | --- | --- |
| **CUSTID** | FNAME1 | LNAME1 | DOB |

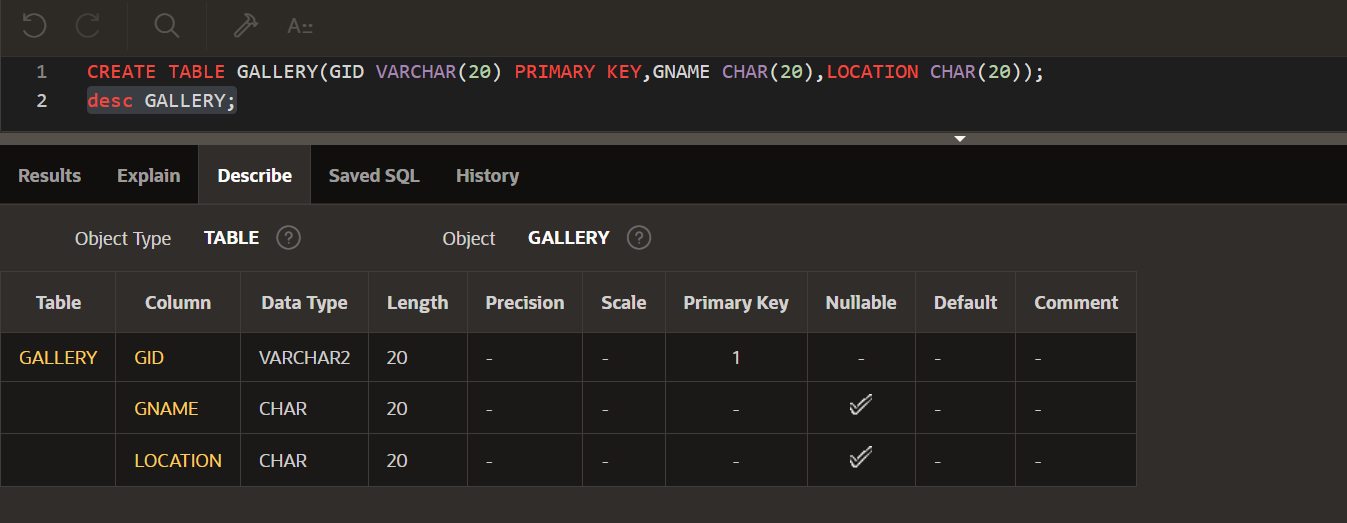
1. **Third Normal Form(3NF):**

A table is in 3NF if it is in 2NF and if it has no transitive dependency. X->Y, Y->Z, X>Z

According to CODD’s definition a relation schema R is in 3NF. It satisfies 2NF and no non-prime attribute of R is transitively dependent on the primary key. All tables of database satisfy upto 3NF.

**1.5 CREATION OF TABLES**

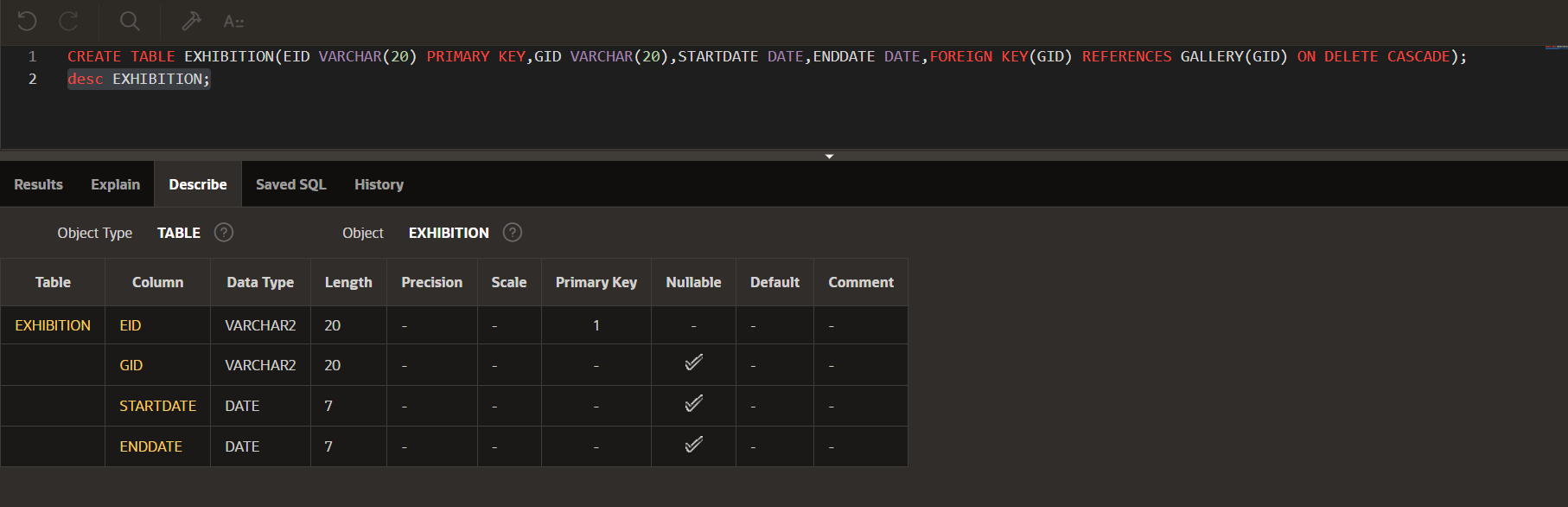
**1. CREATING GALLERY TABLE**

CREATE TABLE GALLERY(GID VARCHAR(20) PRIMARY KEY,GNAME CHAR(20),  
LOCATION CHAR(20));

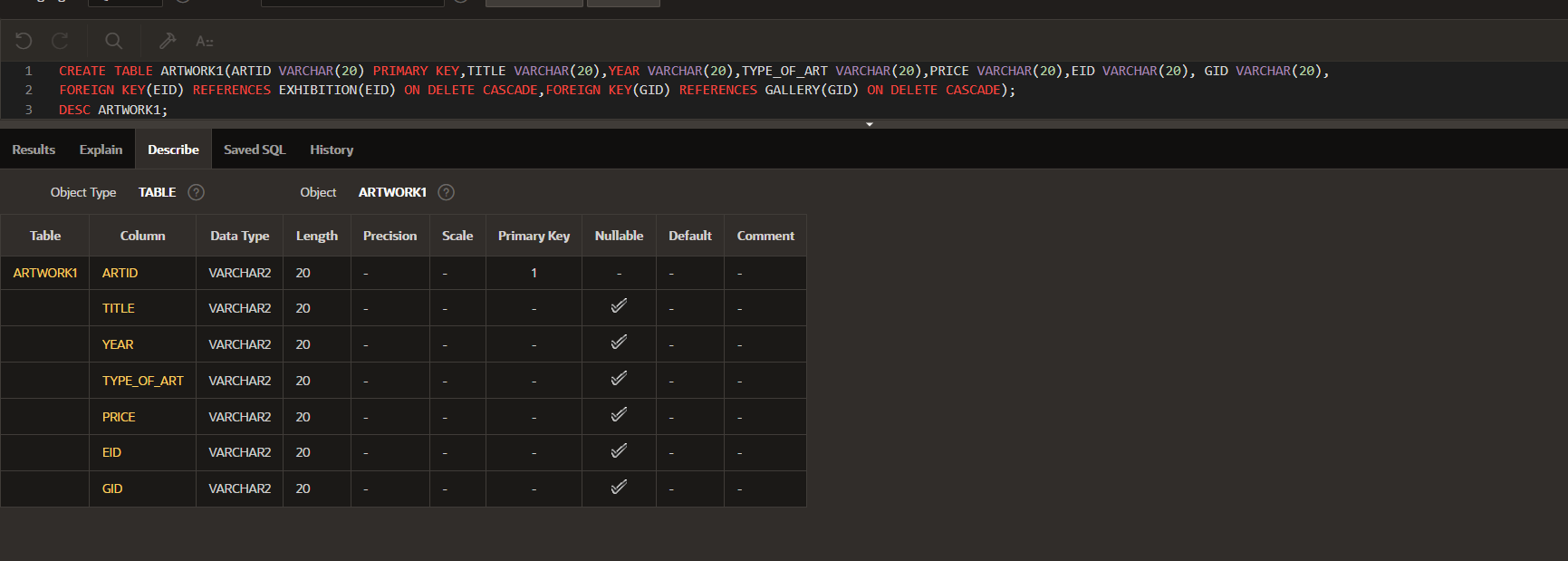
**2. CREATE EXHIBITION TABLE**

CREATE TABLE EXHIBITION(EID VARCHAR(20) PRIMARY KEY,GID VARCHAR(20),  
STARTDATE DATE,ENDDATE DATE,  
FOREIGN KEY(GID) REFERENCES GALLERY(GID) ON DELETE CASCADE);

Use the ON DELETE CASCADE option to specify whether you want rows deleted in a child table when corresponding rows are deleted in the parent table

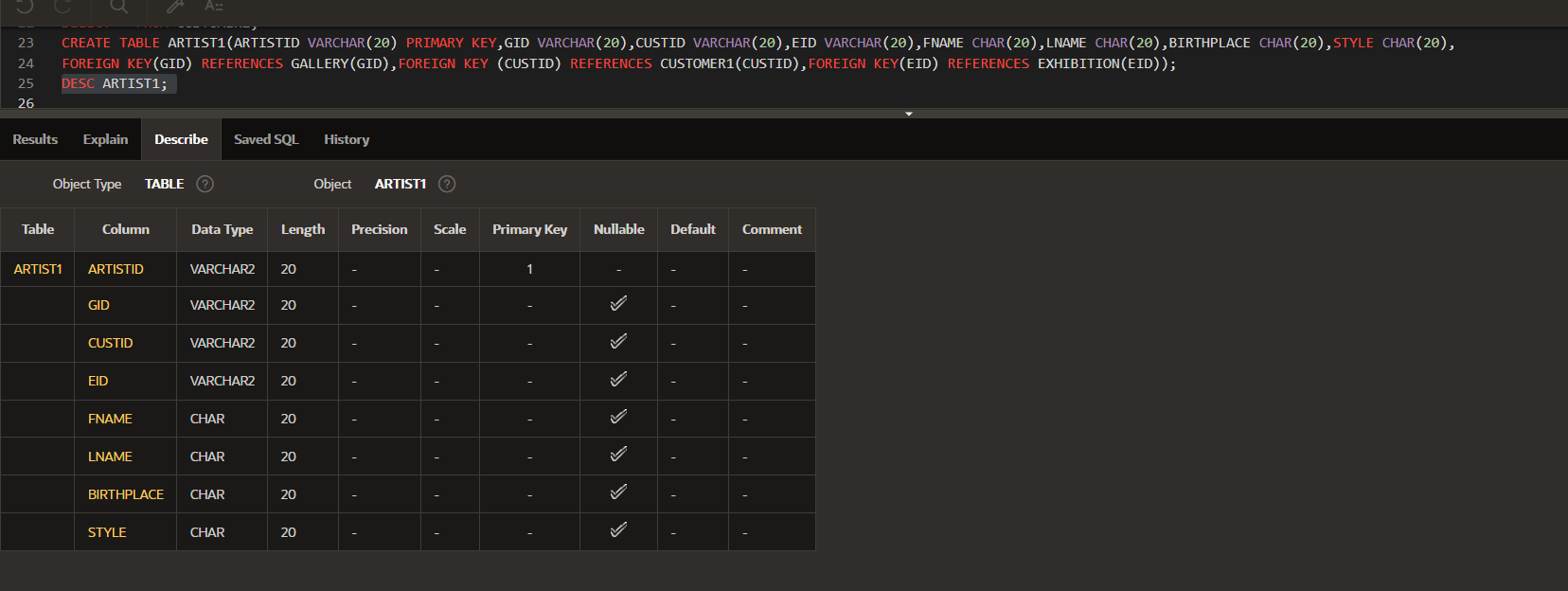


**3.CREATE ARTWORK1 TABLE**

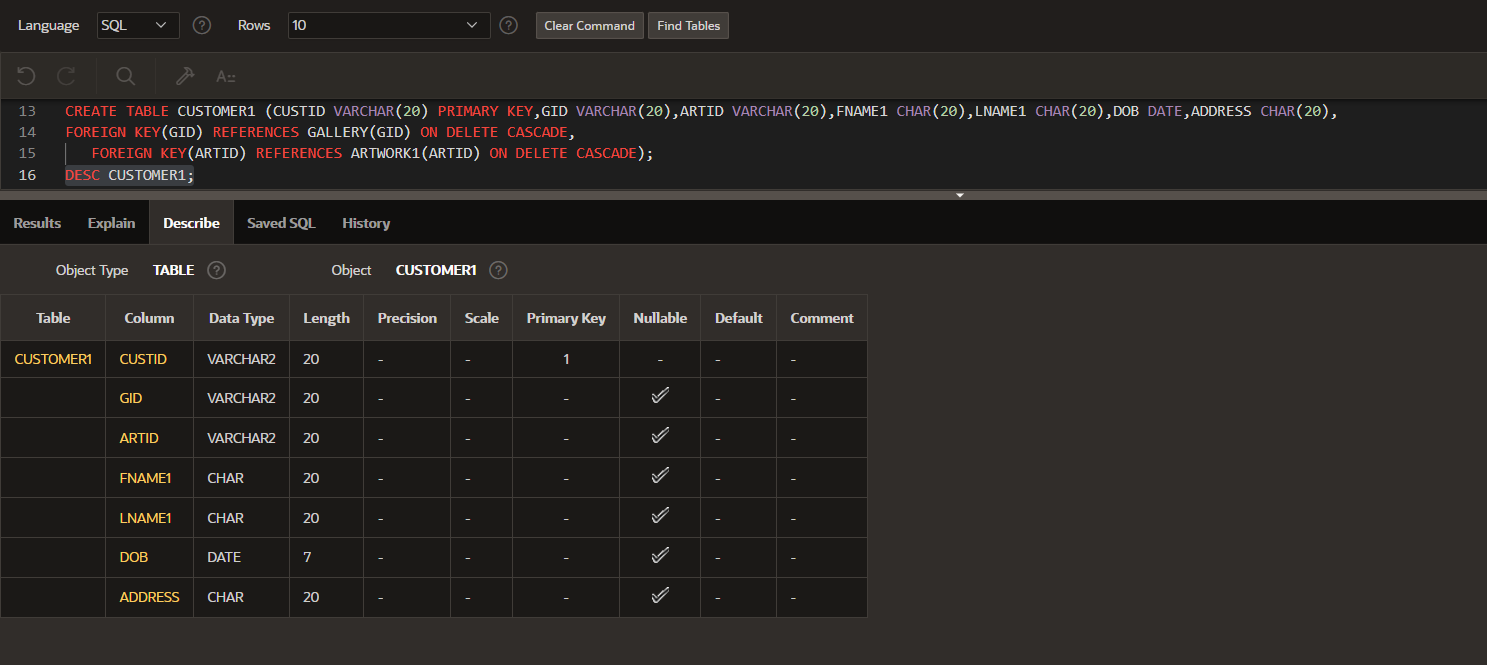
CREATE TABLE ARTWORK1(ARTID VARCHAR(20) PRIMARY KEY,TITLE VARCHAR(20),  
 YEAR VARCHAR(20),TYPE\_OF\_ART VARCHAR(20),PRICE VARCHAR(20),  
 EID VARCHAR(20), GID VARCHAR(20),  
 FOREIGN KEY(EID) REFERENCES EXHIBITION(EID) ON DELETE CASCADE,  
 FOREIGN KEY(GID) REFERENCES GALLERY(GID) ON DELETE CASCADE);

1. **CREATE ARTIST1 TABLE**

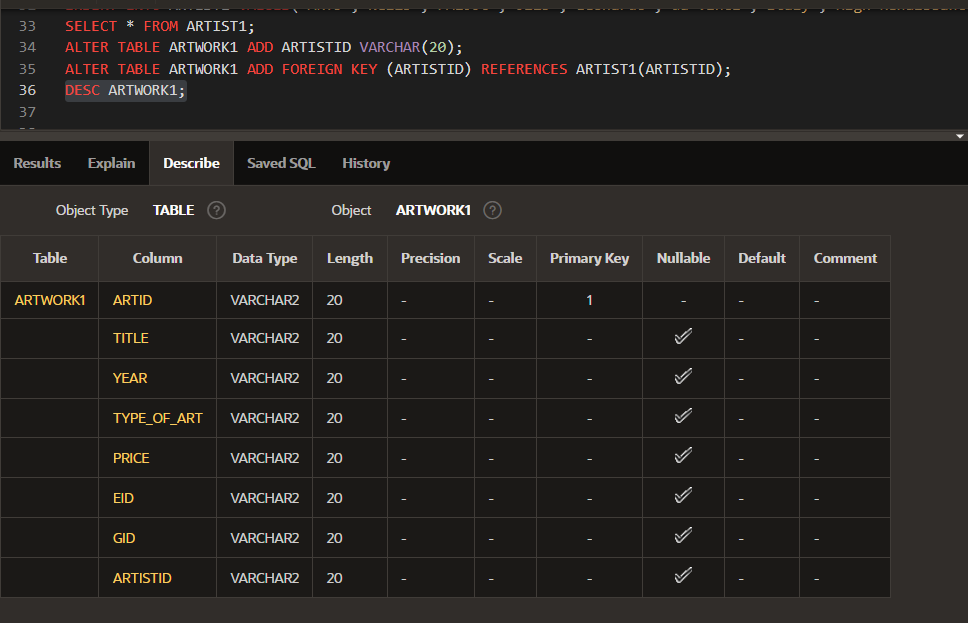
CREATE TABLE ARTIST1 (ARTISTID VARCHAR(20) PRIMARY KEY,GID VARCHAR(20),  
CUSTID VARCHAR(20),EID VARCHAR(20),FNAME CHAR(20),LNAME CHAR(20),  
BIRTHPLACE CHAR(20),STYLE CHAR(20),  
 FOREIGN KEY(GID) REFERENCES GALLERY(GID) ON DELETE CASCADE,  
 FOREIGN KEY (CUSTID) REFERENCES CUSTOMER1(CUSTID) ON DELETE CASCADE,  
 FOREIGN KEY(EID) REFERENCES EXHIBITION(EID) ON DELETE CASCADE);



**5. CREATE CUSTOMER1 TABLE**

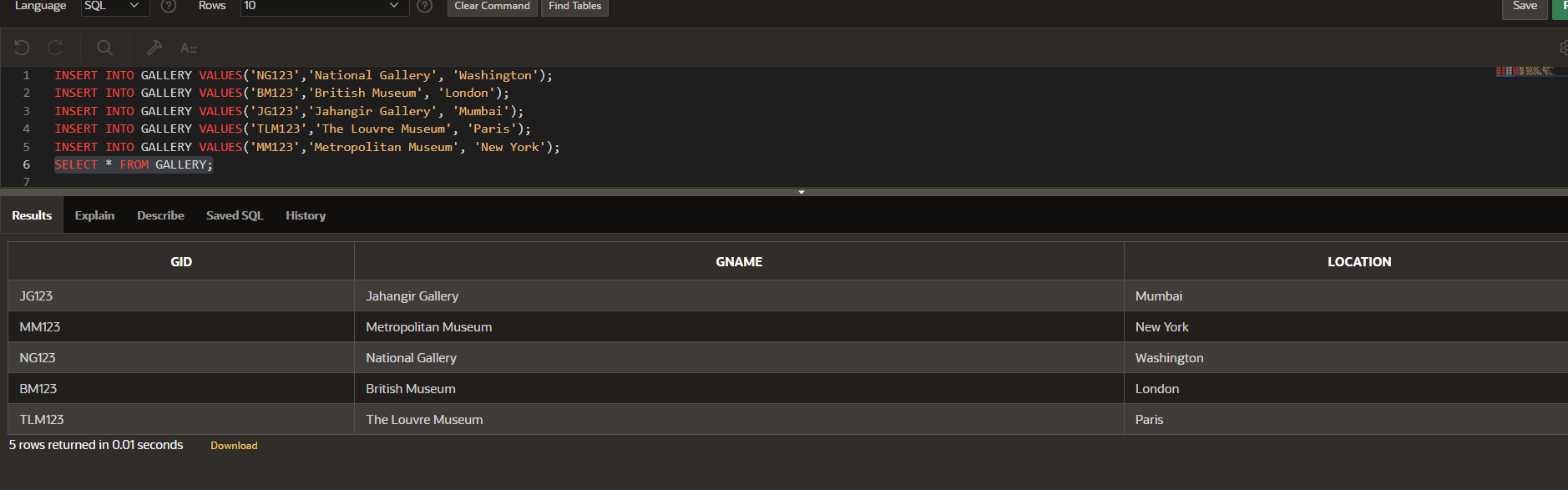
CREATE TABLE CUSTOMER1 (CUSTID VARCHAR(20) PRIMARY KEY,GID VARCHAR(20),  
ARTID VARCHAR(20),FNAME1 CHAR(20),LNAME1 CHAR(20),DOB DATE,ADDRESS CHAR(20),  
FOREIGN KEY(GID) REFERENCES GALLERY(GID) ON DELETE CASCADE,  
 FOREIGN KEY(ARTID) REFERENCES ARTWORK1(ARTID) ON DELETE CASCADE);

**UPDATING ARTWORK TABLE ;**ALTER TABLE ARTWORK1 ADD ARTISTID VARCHAR(20);  
  
 ALTER TABLE ARTWORK1 ADD FOREIGN KEY (ARTISTID) REFERENCES ARTIST1(ARTISTID) ON DELETE CASCADE;

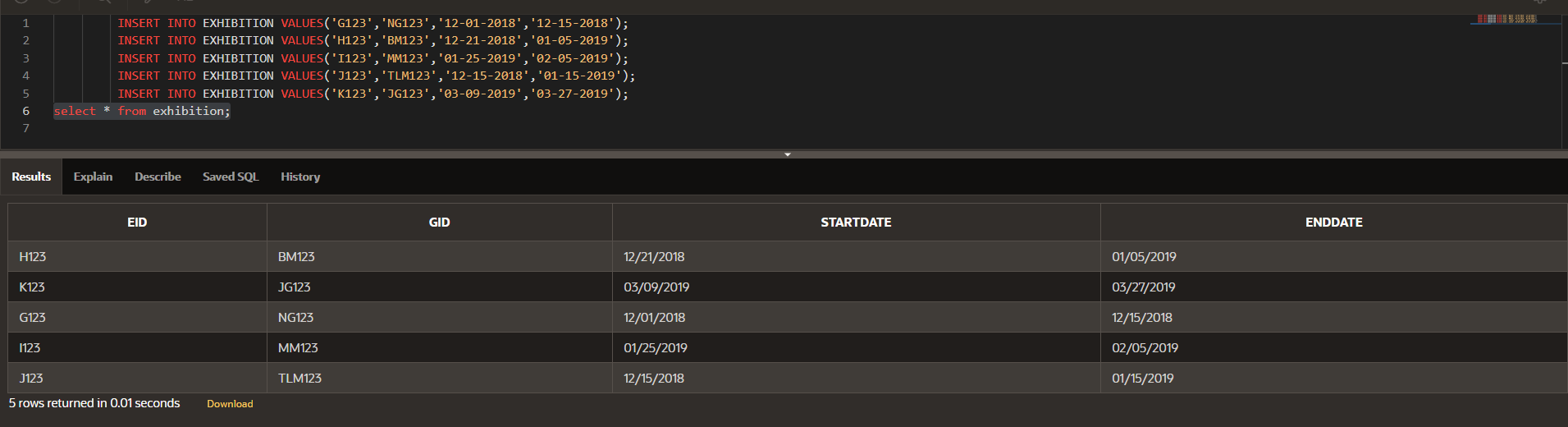


**1.6 INSERTION OF TUPLES**

**1. INSERTION OF GALLERY TABLE**

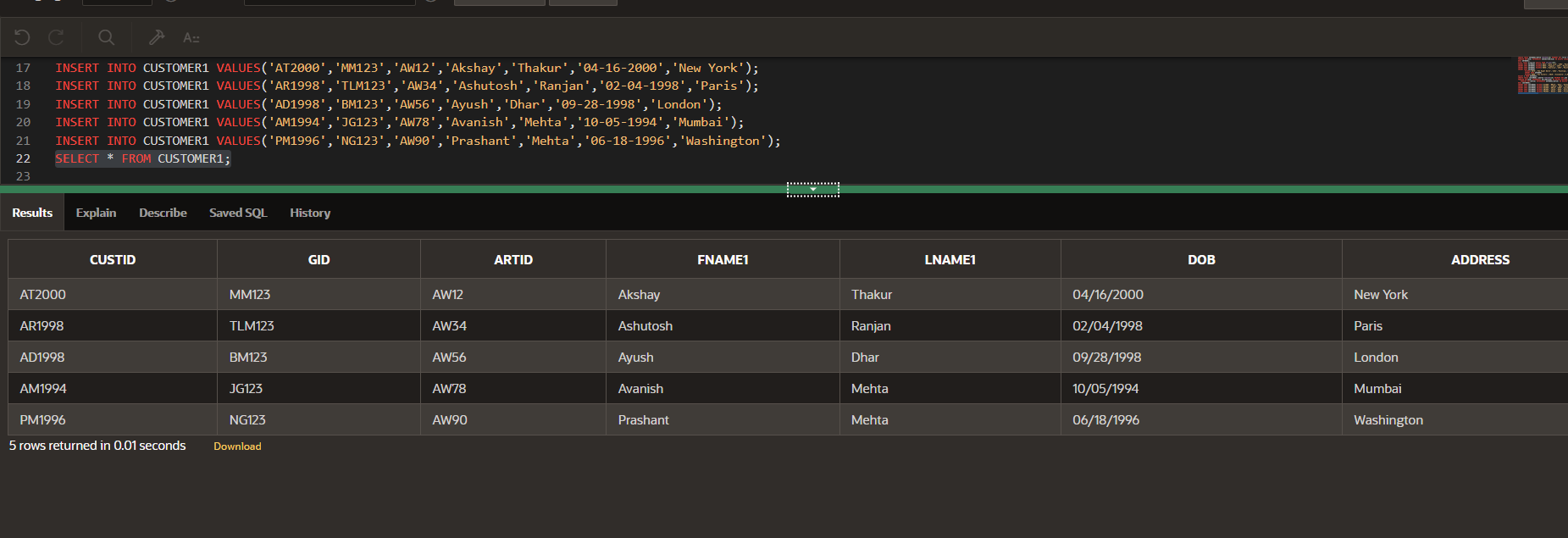
 INSERT INTO GALLERY VALUES('NG123','National Gallery', 'Washington');  
 INSERT INTO GALLERY VALUES('BM123','British Museum', 'London');  
 INSERT INTO GALLERY VALUES('JG123','Jahangir Gallery', 'Mumbai');  
 INSERT INTO GALLERY VALUES('TLM123','The Louvre Museum', 'Paris');  
 INSERT INTO GALLERY VALUES('MM123','Metropolitan Museum', 'New York');

**2. INSERTION OF EXHIBITION TABLE**

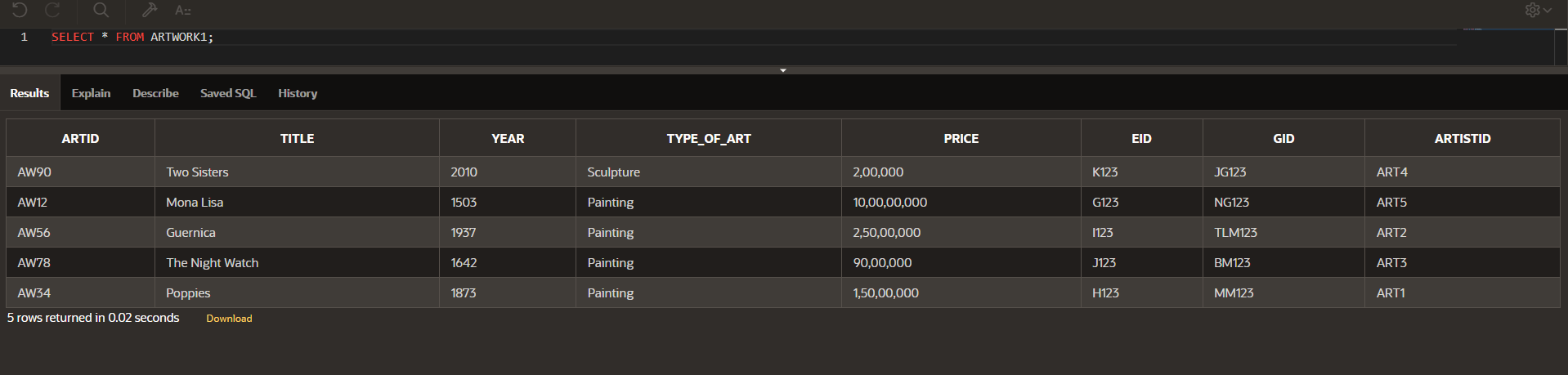
 INSERT INTO EXHIBITION VALUES('G123','NG123','12-01-2018','12-15-2018');  
 INSERT INTO EXHIBITION VALUES('H123','BM123','12-21-2018','01-05-2019');  
 INSERT INTO EXHIBITION VALUES('I123','MM123','01-25-2019','02-05-2019');  
 INSERT INTO EXHIBITION VALUES('J123','TLM123','12-15-2018','01-15-2019');  
 INSERT INTO EXHIBITION VALUES('K123','JG123','03-09-2019','03-27-2019');

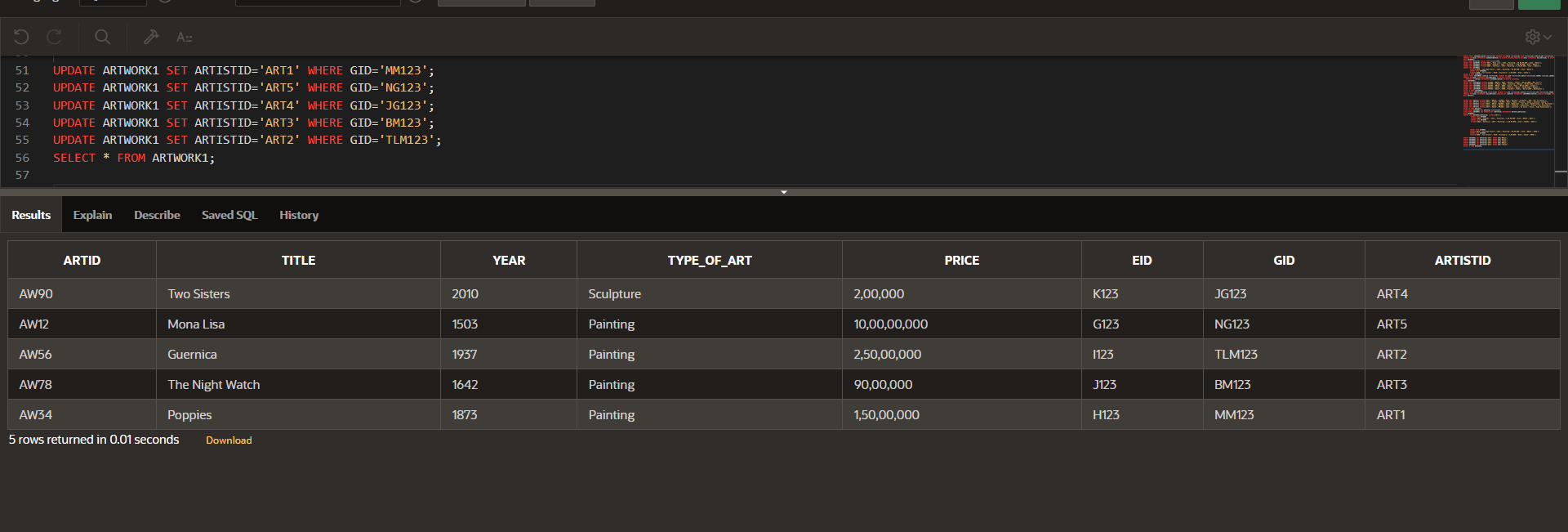
**3.** **INSERTION OF CUSTOMER1 TABLE**

INSERT INTO CUSTOMER1 VALUES('AT2000','MM123','AW12','Akshay','Thakur','04-16-2000','New York');  
 INSERT INTO CUSTOMER1 VALUES('AR1998','TLM123','AW34','Ashutosh','Ranjan','02-04-1998','Paris');  
INSERT INTO CUSTOMER1 VALUES('AD1998','BM123','AW56','Ayush','Dhar','09-28-1998','London');  
INSERT INTO CUSTOMER1 VALUES('AM1994','JG123','AW78','Avanish','Mehta','10-05-1994','Mumbai');  
 INSERT INTO CUSTOMER1 VALUES('PM1996','NG123','AW90','Prashant','Mehta','06-18-1996','Washington');



**4.** **INSERTION OF ARTWORK1 TABLE**

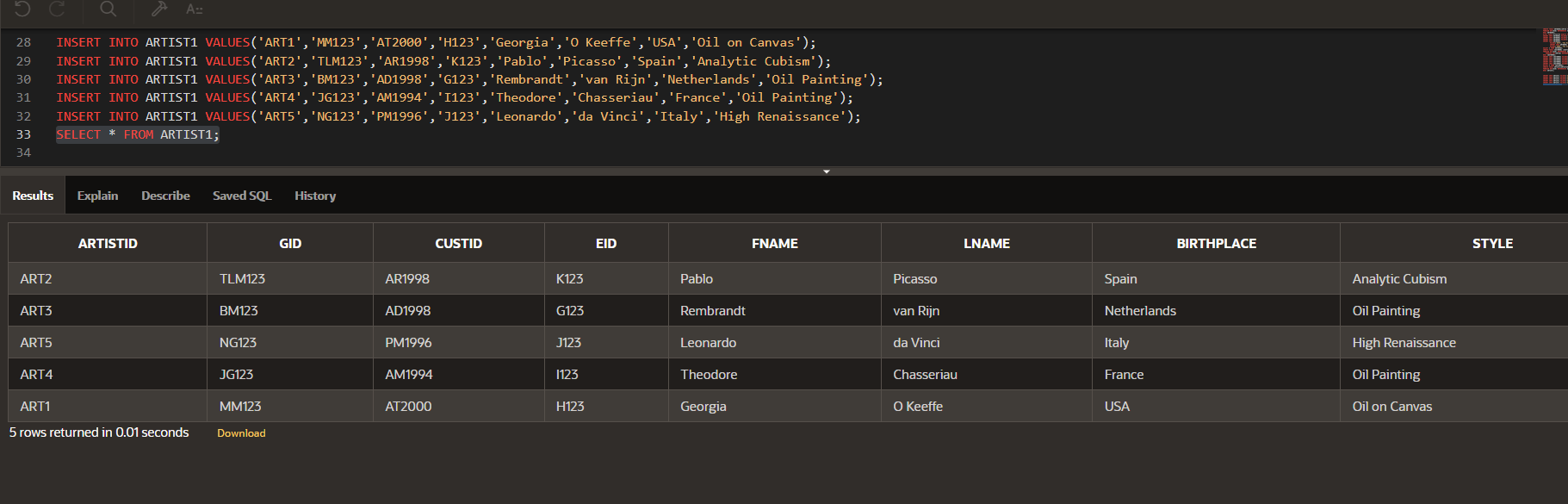
 INSERT INTO ARTWORK1 VALUES('AW12','Mona Lisa','1503',' Painting','10,00,00,000','G123','NG123','ART5');  
 INSERT INTO ARTWORK1 VALUES('AW34','Poppies','1873','Painting','1,50,00,000','H123','MM123','ART1');  
 INSERT INTO ARTWORK1  
 VALUES('AW56','Guernica','1937','Painting','2,50,00,000','I123','TLM123','ART2');  
 INSERT INTO ARTWORK  
 VALUES('AW78','The Night Watch','1642','Painting','90,00,000','J123','BM123','ART3');  
 INSERT INTO ARTWORK  
 VALUES('AW90','Two Sisters','2010','Sculpture','2,00,000','K123','JG123','ART4');

**UPDATING THE VALUES OF ARTISTID1 IN ARTWORK1 TABLE**

**5.** **INSERTION OF ARTIST TABLE**

INSERT INTO ARTIST  
 VALUES('ART1','MM123','AT2000','H123','Georgia','O Keeffe','USA','Oil on Canvas');  
 INSERT INTO ARTIST  
 VALUES('ART2','TLM123','AR1998','K123','Pablo','Picasso','Spain','Analytic Cubism');  
 INSERT INTO ARTIST VALUES

('ART3','BM123','AD1998','G123,'Rembrandt','van Rijn','Netherlands','Oil Painting');

 INSERT INTO ARTIST  
 VALUES('ART4','JG123','AM1994','I123','Theodore','Chasseriau','France','Oil Painting');  
 INSERT INTO ARTIST  
 VALUES('ART5','NG123','PM1996','J123','Leonardo','da Vinci','Italy','High Renaissance');