University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: 2

Date: June 11, 2023

Group Number: 4

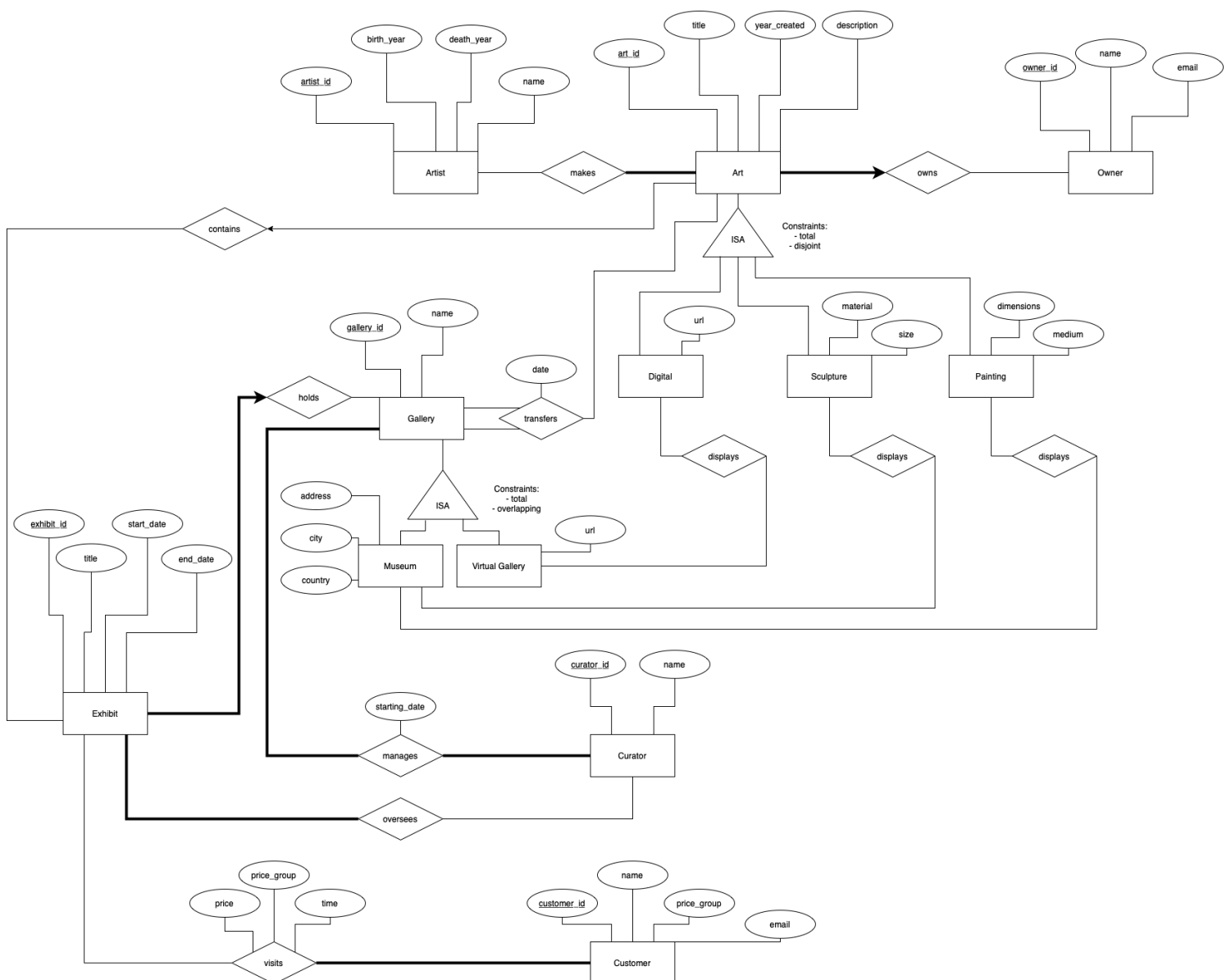| Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Alexander Tsao	77721322	i3l1j	alexanderntsao@gmail.com
Jiyoung Jang	41283839	b7v1q	gsdmlgla@gmail.com
Willis Cao	30939573	v0l1b	wcao2012@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

Project Summary:

Our database serves as a platform for art galleries to oversee logistical information by modeling the transfer of art pieces between locations, the ownership of the art pieces, and the relationships between artists, artwork owners, and gallery curators. The application will be shared and used by art galleries, which may be museums or virtual galleries, worldwide to keep track of the movement of works of art for specific exhibitions. Other information including customers and the fares they pay to enter exhibits will be stored as well.

ER Diagram:

Changes:
- We removed the weak entity set "Fare" because our previous design for that entity set would have included many duplicate values
- Since a customer must "pay" a fare to "visit" an exhibit, we combined these relationship sets into one, since the information stored in each table would essentially be the same
- We also made "Customer" have total participation in the "visits" relationship because every customer must visit some exhibit or else they wouldn't be a customer
- We also added "email" as an attribute in the entity sets "owner" and "customer" in order to include candidate keys
- "url" was added as an attribute to "Virtual Gallery" to mirror how "Museum" has an address as an attribute
- We removed the relationship "interacts_with" between "Owner" and "Curator" because there was nothing significant for the database to keep track of in this relationship

Schema:

Artist (<u>artist_id</u>: integer, birth_year: integer, death_year: integer, name: char(30))
- PK: artist_id

Makes (**<u>artist_id</u>**: integer, **<u>art_id</u>**: integer)
- PK: {artist_id, art_id}
- FK: artist_id, art_id
- Constraints: artist_id NOT NULL

Art (<u>art_id</u>: integer, title: char(30), year_created: integer, description: char(1000), **owner_id**: integer, **exhibit_id**: integer)
- PK: art_id
- FK: owner_id, exhibit_id
- Constraints: owner_id NOT NULL

Owner (<u>owner_id</u>: integer, name: char(30), email: char(50))
- PK: owner_id
- CK: email
- Constraints: email UNIQUE

Digital (**<u>art_id</u>**: integer, url: char(100))
- PK/FK: art_id

Sculpture (**<u>art_id</u>**: integer, material: char(20), size: number(10, 2))
- PK/FK: art_id

Painting (**<u>art_id</u>**: integer, dimensions: char(20), medium: char(20))
- PK/FK: art_id

Displays (**art_id**: integer, **gallery_id**: integer)
- PK: {art_id, gallery_id}
- FK: art_id, gallery_id

Gallery (gallery_id: integer, name: char(30))
- PK: gallery_id

Museum (**gallery_id**: integer, address: char(100), city: char(50), country: char(30))
- PK/FK: gallery_id

Virtual_Gallery (**gallery_id**: integer, url: char(100))
- PK/FK: gallery_id

Curator (curator_id: integer, name: char(30))
- PK: curator_id

Manages (**curator_id**: integer, **gallery_id**: integer, starting_date: date)
- PK: {curator_id, gallery_id}
- FK: curator_id, gallery_id

Oversees (**curator_id**: integer, **exhibit_id**: integer)
- PK: {curator_id, exhibit_id}
- FK: curator_id, exhibit_id

Exhibit (exhibit_id: integer, title: char(30), starting_date: date, ending_date: date, **gallery_id**: integer)
- PK: exhibit_id
- FK: gallery_id
- Constraints: gallery_id NOT NULL

Transfers (**art_id**: integer, **origin**: integer, **destination**: integer, date: date)
- PK: {art_id, origin, destination}
- FK: art_id, origin, destination
- Notes:
    - Transfer is the relationship set "transfers_art_to" from the ER diagram
    - Origin is the gallery_id where the art piece comes from
    - Destination is the gallery_id where the art piece goes to

Customer (customer_id: integer, name: char(30), price_group: char(20), email: char(50))
- PK: customer_id
- CK: email
- Constraints: email UNIQUE, price_group NOT NULL, email NOT NULL

Visits (**customer_id**: integer, **exhibit_id**: integer, time: timestamp, price: integer, price_group: char(20)
- PK: {customer_id, exhibit_id}
- FK: customer_id, exhibit_id
- CK: {customer_id, time}
- Constraints: exhibit_id NOT NULL, time NOT NULL

Functional Dependencies:

not identified by PK
Art.art_id → Art.owner_id
Art.art_id → Art.exhibit_id
Exhibit.exhibit_id → Exhibit.gallery_id
Visits.customer_id, Visits.time → Visits.exhibit_id
Visits.exhibit_id, Visits.price → Visits.price_group

Normalization:

Normalizing the "Visits" relationship
Visits (customer_id, exhibit_id, time, price, price_group)

FDs:
not identified by PK
customer_id, time → exhibit_id
exhibit_id, price → price_group
customer_id, exhibit_id → time, price, price_group

Step 1: Find the minimal key

Left	Middle	Right
customer_id	time, exhibit_id, price	price_group

customer_id, time+ = {customer_id, exhibit_id, time, price, price_group} → key!
customer_id, exhibit_id+ = {customer_id, exhibit_id, time, price, price_group} → key!

Step 2: Find the minimal cover
   a. Standard form
       customer_id, time → exhibit_id
       exhibit_id, price → price_group
       customer_id, exhibit_id → time

customer_id, exhibit_id → price
customer_id, exhibit_id → price_group
   b. Simplify LHS
       Nothing to simplify.
   c. Delete redundant FDs
       The FD customer_id, exhibit_id → price_group is redundant, so it can be removed.

       Minimal cover:
       customer_id, time → exhibit_id
       exhibit_id, price → price_group
       customer_id, exhibit_id → time
       customer_id, exhibit_id → price

Step 3: Decomposition into 3NF using synthesis method

R1 (**customer_id**, time, **exhibit_id**)
R2 (**exhibit_id**, price, price_group)
R3 (**customer_id, exhibit_id**, **time**, price)

Q: Are either of the minimal keys included in one of these above relationships?
A: Yes, both R1 and R3. No need to add relations.

Step 4: Tables Post-Normalization:

CustomerVisitsExhibit (**customer_id**: integer, time: timestamp, **exhibit_id**: integer)
   - PK: {customer_id, time}
   - CK: {customer_id, exhibit_id}
   - FK: customer_id, exhibit_id
   - Constraints: time NOT NULL
ExhibitCharges (**exhibit_id**: integer, price: integer, price_group: char(20))
   - PK: {exhibit_id, price}
   - FK: exhibit_id
Visits (**customer_id**: integer, **exhibit_id**: integer, **time**: timestamp, **price**: integer)
   - PK: {customer_id, exhibit_id}
   - CK: {customer_id, time}
   - FK: customer_id, exhibit_id, time, price

SQL DDL Statements:

CREATE TABLE Artist (
        artist_id INT,
        birth_year INT,
        death_year INT,

```sql
        name VARCHAR2(30),
        PRIMARY KEY (artist_id)
);

CREATE TABLE Owner (
        owner_id        INT,
        name VARCHAR2(30)
        PRIMARY KEY (owner_id)
);

CREATE TABLE Customer (
        customer_id    INT,
        name            VARCHAR2(30),
        price_group    VARCHAR2(20) NOT NULL,
        email            VARCHAR2(50) NOT NULL,
        PRIMARY KEY (customer_id),
        CONSTRAINT Customer UNIQUE (email)
);

CREATE TABLE Gallery (
        gallery_id INT,
        name VARCHAR2(30),
        PRIMARY KEY (gallery_id)
);

CREATE TABLE Curator (
        curator_id      INT,
        name            VARCHAR2(30),
        PRIMARY KEY (curator_id)
);

CREATE TABLE Art (
        art_id INT,
        owner_id INT NOT NULL,
        title VARCHAR2(30),
        year_created INT,
        description VARCHAR2(4000)
        exhibit_id INT,
        PRIMARY KEY (art_id),
        FOREIGN KEY (owner_id) REFERENCES Owner (owner_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE,
        FOREIGN KEY (exhibit_id) REFERENCES Exhibit (exhibit_id),
);
```

```sql
CREATE TABLE Makes (
        art_id INT,
        artist_id INT,
        PRIMARY KEY (art_id, artist_id),
        FOREIGN KEY (art_id) REFERENCES Art (art_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE,
        FOREIGN KEY (artist_id) REFERENCES Artist (artist_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE,
);

CREATE Displays (
        art_id INT,
        gallery_id INT,
        PRIMARY KEY (art_id, gallery_id),
        FOREIGN KEY (art_id) REFERENCES Art (art_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE,
        FOREIGN KEY (gallery_id) REFERENCES Gallery (gallery_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE
);

CREATE TABLE Digital (
        art_id INT,
        url VARCHAR2(100),
        PRIMARY KEY (art_id),
        FOREIGN KEY (art_id) REFERENCES Art (art_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE
);

CREATE TABLE Sculpture (
        art_id INT,
        material VARCHAR2(20),
        size NUMBER(10, 2),
        PRIMARY KEY (art_id),
        FOREIGN KEY (art_id) REFERENCES Art (art_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE
);
```

```sql
CREATE TABLE Painting (
        art_id INT,
        dimension VARCHAR2(20),
        medium VARCHAR2(20)
        PRIMARY KEY (art_id),
        FOREIGN KEY (art_id) REFERENCES Art (art_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE
);

CREATE TABLE Manages (
        curator_id      INT,
        gallery_id      INT,
        starting_date   DATE,
        PRIMARY KEY (curator_id, gallery_id),
        FOREIGN KEY (curator_id)
                REFERENCES Curator (curator_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE,
        FOREIGN KEY (gallery_id)
                REFERENCES Gallery (gallery_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE
);

CREATE TABLE Oversees (
        curator_id      INT,
        exhibit_id      INT,
        PRIMARY KEY (curator_id, exhibit_id),
        FOREIGN KEY (curator_id)
                REFERENCES Curator (curator_id),
                ON UPDATE CASCADE
                ON DELETE CASCADE,
        FOREIGN KEY (exhibit_id)
                REFERENCES Exhibit (exhibit_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE
);

CREATE TABLE Museum (
        gallery_id      INT,
        address         VARCHAR2(100),
        city            VARCHAR2(50),
```

```sql
        country        VARCHAR2(30),
        PRIMARY KEY (gallery_id),
        FOREIGN KEY (gallery_id ) REFERENCES Art (gallery_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE
);

CREATE TABLE Virtual_Gallery (
        gallery_id      INT,
        url             VARCHAR2(100),
        PRIMARY KEY (gallery_id),
        FOREIGN KEY (gallery_id ) REFERENCES Art (gallery_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE
);

CREATE TABLE Transfers (
        art_id                  INT,
        origin                  INT,
        date                    DATE,
        destination             INT,
        PRIMARY KEY (art_id, origin, destination),
        FOREIGN KEY(art_id)
                REFERENCES Art (art_id)
                ON UPDATE CASCADE,
        FOREIGN KEY(origin, destination)
                REFERENCES Gallery(gallery_id, gallery_id)
                ON DELETE SET NULL,
);

CREATE TABLE Exhibit (
        exhibit_id INT,
        gallery_id INT,
        title VARCHAR2(30),
        start_date DATE,
        end_date DATE,
        PRIMARY KEY (exhibit_id)
        FOREIGN KEY (gallery_id)
                REFERENCES Gallery (gallery_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE
);
```

```sql
CREATE TABLE Visits (
        customer_id    INT,
        exhibit_id     INT,
        time           timestamp NOT NULL,
        price          INT,
        PRIMARY KEY (customer_id, exhibit_id),
        FOREIGN KEY (customer_id, time)
                REFERENCES CustomerVisitsExhibit (customer_id, time)
                ON UPDATE CASCADE
                ON DELETE CASCADE,
        FOREIGN KEY (price)
                REFERENCES ExhibitCharges (price),
        FOREIGN KEY (exhibit_id)
                REFERENCES Exhibit (exhibit_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE
);

CREATE TABLE CustomerVisitsExhibit (
        customer_id    INT,
        time           TIMESTAMP,
        exhibit_id     INT,
        PRIMARY KEY (customer_id, time),
        FOREIGN KEY(customer_id)
                REFERENCES Customer(customer_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE,
        FOREIGN KEY(exhibit_id)
                REFERENCES Exhibit (exhibit_id)
                ON UPDATE CASCADE
                ON DELETE CASCADE,
);

CREATE TABLE ExhibitCharges (
        exhibit_id     INT,
        price          INT,
        price_group    VARCHAR2 (20),
        PRIMARY KEY (exhibit_id, price),
        FOREIGN KEY (exhibit_id)
                REFERENCES Exhibit (id)
                ON UPDATE CASCADE
                ON DELETE CASCADE
);
```

Insert Statements

INSERT INTO Gallery (gallery_id, name)
VALUES(1, "Vancouver Art Gallery");

INSERT INTO Gallery (gallery_id, name)
VALUES(2, "The British Museum");

INSERT INTO Gallery (gallery_id, name)
VALUES(3, "Canadian Online Art Library");

INSERT INTO Gallery (gallery_id, name)
VALUES(4, "Los Angeles County Museum of Art");

INSERT INTO Gallery (gallery_id, name)
VALUES(5, "Free Digital Art Gallery");

INSERT INTO Gallery (gallery_id, name)
VALUES(6, "National Museum of Modern and Contemporary Art");

INSERT INTO Gallery (gallery_id, name)
VALUES(7, "China Art Museum");

INSERT INTO Gallery (gallery_id, name)
VALUES(8, "SpaceArtCollection.com");

INSERT INTO Gallery (gallery_id, name)
VALUES(9, "Bob's Gallery");

INSERT INTO Customer (customer_id,name,price_group,email)
VALUES (1, "John Johnson", "Adult", "JohnJohnson@gmail.com");

INSERT INTO Customer (customer_id,name,price_group,email)
VALUES (2, "Frank Sausage", "Adult", "SausageFrank@mail.com");

INSERT INTO Customer (customer_id,name,price_group,email)
VALUES (3, "Sarah Kerrigan", "Student", "heartsoftheswarm@gmail.com");

INSERT INTO Customer (customer_id,name,price_group,email)
VALUES (4, "Dekerd Cain", "Senior", "diablooldguy@gmail.com");

INSERT INTO Customer (customer_id,name,price_group,email)
VALUES (5, "Dale Falconer", "Adult", "dalefalconer69@gmail.com");

```sql
INSERT INTO Artist (artist_id,birth_year,death_year,name)
VALUES(1, 1881, 1973, "Pablo Picasso");

INSERT INTO Artist (artist_id,birth_year,death_year,name)
VALUES(2, 1452, 1519, "Leonardo da Vinci");

INSERT INTO Artist (artist_id,birth_year,death_year,name)
VALUES(3, 1871, 1945, "Emily Carr");

INSERT INTO Artist (artist_id,birth_year,death_year,name)
VALUES(4, 1946, NULL, "Stephen Lack");

INSERT INTO Artist (artist_id,birth_year,death_year,name)
VALUES(5, 1941, NULL, "Charlie Inukpuk");

INSERT INTO Artist (artist_id,birth_year,death_year,name)
VALUES(6, 2010, 2023, "Jay");

INSERT INTO Artist (artist_id,birth_year,death_year,name)
VALUES(7, 1995, NULL, "Sarah Smith");

INSERT INTO Curator(curator_id, name)
VALUES(1, "Joseph Joestar");

INSERT INTO Curator(curator_id, name)
VALUES(2, "Maruf Aristeides");

INSERT INTO Curator(curator_id, name)
VALUES(3, "Ventsislav Zakaria");

INSERT INTO Curator(curator_id, name)
VALUES(4, "Thorsten Sudhir");

INSERT INTO Curator(curator_id, name)
VALUES(5, "Roddy Włodzimierz");

INSERT INTO Owner (owner_id, name)
VALUES(1, "British Museum");

INSERT INTO Owner (owner_id, name)
VALUES(2, "Bob Richman");
```

```sql
INSERT INTO Owner (owner_id, name)
VALUES(3, "Canadian Art Society");

INSERT INTO Owner (owner_id, name)
VALUES(4, "Indigenous Art Gallery");

INSERT INTO Owner (owner_id, name)
VALUES(5, "Vancouver Art Gallery");

INSERT INTO Museum (gallery_id, address, city, country)
VALUES(1, "750 Hornby St", "Vancouver", "Canada");

INSERT INTO Museum (gallery_id, address, city, country)
VALUES(2, "Great Russell St, London WC1B 3DG", "London", "The United Kingdom of Great
Britain and Northern Ireland");

INSERT INTO Museum (gallery_id, address, city, country)
VALUES(4, "5905 Wilshire Blvd", "Los Angeles", "USA");

INSERT INTO Museum (gallery_id, address, city, country)
VALUES(6, "30 Samcheong-ro 5-gil, Jongno-gu", "Seoul", "Republic of Korea");

INSERT INTO Museum (gallery_id, address, city, country)
VALUES(7, "205 Shangnan Road", "Shanghai", "China");

INSERT INTO Virtual_Gallery (gallery_id, url)
VALUES(3, "COAL.com");

INSERT INTO Virtual_Gallery (gallery_id, url)
VALUES(5, "FreeDigitalArtGallery.com");

INSERT INTO Virtual_Gallery (gallery_id, url)
VALUES(8, "SpaceArtCollection.com");

INSERT INTO Virtual_Gallery (gallery_id, url)
VALUES(9, "BobsArtHaven.com");

INSERT INTO Virtual_Gallery (gallery_id, url)
VALUES(2, "https://www.britishmuseum.org/collection");

INSERT INTO Exhibit (exhibit_id, gallery_id, title, start_date, end_date)
VALUES(1, 2, "2025 British Museum Back to the Past", "2025-04-01", "2025-04-06");
```

INSERT INTO Exhibit (exhibit_id, gallery_id, title, start_date, end_date)
VALUES(2, 1, "2026 Winter Picasso Special", "2026-11-25", "2026-12-25");

INSERT INTO Exhibit (exhibit_id, gallery_id, title, start_date, end_date)
VALUES(3, 6, "Tradition meets VR", "2025-07-02", "2025-07-22");

INSERT INTO Exhibit (exhibit_id, gallery_id, title, start_date, end_date)
VALUES(4, 4, "LA Spring Exhibit", "2027-03-01", "2027-04-30");

INSERT INTO Exhibit (exhibit_id, gallery_id, title, start_date, end_date)
VALUES(5, 7, "Meet the Emperor Exhibit", "2026-06-01", "2026-09-01");

INSERT INTO Art (art_id,owner_id,title,year_created,description, exhibit_id)
VALUES(1, 1, "Guernica", 1937, "The grey, black, and white painting, on a canvas 3.49 meters
(11 ft 5 in) tall and 7.76 meters (25 ft 6 in) across, portrays the suffering wrought by violence and
chaos. Prominent in the composition are a gored horse, a bull, screaming women, a dead baby,
a dismembered soldier, and flames.", NULL);

INSERT INTO Art (art_id,owner_id,title,year_created,description, exhibit_id)
VALUES(2, 2, "The Last Supper", 1498, "The painting represents the scene of the Last Supper
of Jesus with the Twelve Apostles, as it is told in the Gospel of John – specifically the moment
after Jesus announces that one of his apostles will betray him. Its handling of space, mastery of
perspective, treatment of motion and complex display of human emotion has made it one of the
Western world's most recognizable paintings and among Leonardo's most celebrated works.
Some commentators consider it pivotal in inaugurating the transition into what is now termed the
High Renaissance.", NULL);

INSERT INTO Art (art_id,owner_id,title,year_created,description, exhibit_id)
VALUES(3, 3, "The Indian Church", 1929, 'The Indian Church is considered a "transitional"
painting because it reflects the transition of Carr's artistic work from purely depicting Native Art
to shifting her focus toward the land. In her 1946 autobiography, Growing Pains, Carr wrote that
she "felt the subject deeply". She painted it at Friendly Cove, near a lighthouse.', NULL);

INSERT INTO Art (art_id,owner_id,title,year_created,description, exhibit_id)
VALUES(4, 4, "Burnt Kiss", NULL, NULL, NULL);

INSERT INTO Art (art_id,owner_id,title,year_created,description, exhibit_id)
VALUES(5, 5, "UNTITLED - CROUCHING HUNTER", NULL, 'Charlie Inukpuk told Louis
Gagnon: \"When I first started carving, the main challenge was determining the hardness of the
soapstone since the density varies. It was easier to work with the softer stone. I would start by
axe-chipping it, and as I was doing this I would start visualizing it. The shape of the stone also
provided me with ideas. As I axed it gradually, my thoughts would go: \'This is shaping like this...
It is going to be…\' and the carving being formed would follow my thoughts from there. ...We

used shoe polish [to darken the stone] because it made the carvings more attractive and shiny.\"", NULL);

INSERT INTO Art (art_id,owner_id,title,year_created,description, exhibit_id)
VALUES(6, 5, "Man", 1982, NULL, NULL);

INSERT INTO Art (art_id,owner_id,title,year_created,description, exhibit_id)
VALUES(7, 3, "Man with Seal", 1987, NULL, NULL);

INSERT INTO Art (art_id,owner_id,title,year_created,description, exhibit_id)
VALUES(8, 1, "a black stone carving of a hunter with seal", NULL, NULL, NULL);

INSERT INTO Art (art_id,owner_id,title,year_created,description, exhibit_id)
VALUES(9, 2, "Mona Lisa", 1506, "Considered an archetypal masterpiece of the Italian Renaissance, it has been described as \"the best known, the most visited, the most written about, the most sung about, the most parodied work of art in the world\". The painting's novel qualities include the subject's enigmatic expression, monumentality of the composition, the subtle modeling of forms, and the atmospheric illusionism.", NULL);

INSERT INTO Art (art_id,owner_id,title,year_created,description, exhibit_id)
VALUES(10, 3, "The Weeping Woman", 1937, "The Weeping Woman (French: La Femme qui pleure) is a series of oil on canvas paintings by Pablo Picasso, the last of which was created in late 1937. The paintings depict Dora Maar, Picasso\'s mistress and muse. The Weeping Woman paintings were produced by Picasso in response to the bombing of Guernica in the Spanish Civil War and are closely associated with the iconography in his painting Guernica. Picasso was intrigued with the subject of the weeping woman, and revisited the theme numerous times that year. The last version, created on 26 October 1937, was the most elaborate of the series, and has been housed in the collection of the Tate Modern in London since 1987. Another Weeping Woman painting is housed at the National Gallery of Victoria and was involved in a high-profile political art theft.", NULL);

INSERT INTO Art (art_id,owner_id,title,year_created,description, exhibit_id)
VALUES(11, 3, "Alien Invasion", 2021, "Digital creation depicting contact between an unidentified flying object and humankind.", NULL);

INSERT INTO Art (art_id,owner_id,title,year_created,description, exhibit_id)
VALUES(12, 2, "Hula Hoop", 2016, "Kids playing in the sun on a summer day.", NULL);

INSERT INTO Art (art_id,owner_id,title,year_created,description, exhibit_id)
VALUES(13, 4, "Stubbed Toe", 2008, "The pain of squishing a foot against a corner.", NULL);

INSERT INTO Art (art_id,owner_id,title,year_created,description, exhibit_id)
VALUES(14, 2, "Electric Boogaloooo", 2020, "It is a true electric boogaloo 2 when it came out.", NULL);

```sql
INSERT INTO Art (art_id,owner_id,title,year_created,description, exhibit_id)
VALUES(15, 5, "Double Rainbow Triangles", 2022, "Double Rainbow Triangles? What does it mean?", NULL);

INSERT INTO Makes (art_id, artist_id)
VALUES(1, 1);

INSERT INTO Makes (art_id, artist_id)
VALUES(2, 2);

INSERT INTO Makes (art_id, artist_id)
VALUES(3, 3);

INSERT INTO Makes (art_id, artist_id)
VALUES(4, 4);

INSERT INTO Makes (art_id, artist_id)
VALUES(5, 5);

INSERT INTO Makes (art_id, artist_id)
VALUES(6, 5);

INSERT INTO Makes (art_id, artist_id)
VALUES(7, 5);

INSERT INTO Makes (art_id, artist_id)
VALUES(8, 5);

INSERT INTO Makes (art_id, artist_id)
VALUES(9, 2);

INSERT INTO Makes (art_id, artist_id)
VALUES(10, 1);

INSERT INTO Makes (art_id, artist_id)
VALUES(11, 7);
```

```sql
INSERT INTO Makes (art_id, artist_id)
VALUES(12, 7);

INSERT INTO Makes (art_id, artist_id)
VALUES(13, 7);

INSERT INTO Makes (art_id, artist_id)
VALUES(14, 6);

INSERT INTO Makes (art_id, artist_id)
VALUES(15, 6);

INSERT INTO Sculpture (art_id, material, size)
VALUES(4, "Bronze", "24\"x 4\"x 9\"");

INSERT INTO Sculpture (art_id, material, size)
VALUES(5, "Stone", "9.5 X 10 X 6 in.");

INSERT INTO Sculpture (art_id, material, size)
VALUES(6, "Stone", "4⅛ in x 7⅝ in x 3⅝ in");

INSERT INTO Sculpture (art_id, material, size)
VALUES(7, "Stone", "7¾ in x 11⅜ in x 3¾ in");

INSERT INTO Sculpture (art_id, material, size)
VALUES(8, "Stone", "7 x 8 x 7 in. ");

INSERT INTO Digital (art_id, url)
VALUES(11, "sarahsmith.com/alieninvasion");

INSERT INTO Digital (art_id, url)
VALUES(12, "sarahsmith.com/hulahoop");

INSERT INTO Digital (art_id, url)
VALUES(13, "sarahsmith.com/stubbedtoe");

INSERT INTO Digital (art_id, url)
VALUES(14, "jayart/electricboogallooooo.com");

INSERT INTO Digital (art_id, url)
VALUES(15, "jayart/doublerainbowtriangle.com");
```

INSERT INTO Painting (art_id, dimension, medium)
VALUES(1, "137.4 in × 305.5 in", "Oil on canvas");

INSERT INTO Painting (art_id, dimension, medium)
VALUES(2, "181 in × 346 in", "Tempera on gesso, pitch, and mastic");

INSERT INTO Painting (art_id, dimension, medium)
VALUES(3, "42.8 in × 27.1 in", "Oil on canvas");

INSERT INTO Painting (art_id, dimension, medium)
VALUES(9, "30 in × 21 in", "Oil on poplar panel");

INSERT INTO Painting (art_id, dimension, medium)
VALUES(10, "23 15/60 in × 19 11/16 in", "Oil on canvas");

INSERT INTO Manages (curator_id, gallery_id, starting_date)
VALUES(1, 1, "2013-11-15");

INSERT INTO Manages (curator_id, gallery_id, starting_date)
VALUES(2, 2, "1999-04-31");

INSERT INTO Manages (curator_id, gallery_id, starting_date)
VALUES(3, 3, "2000-02-31");

INSERT INTO Manages (curator_id, gallery_id, starting_date)
VALUES(4, 4, "1998-12-25");

INSERT INTO Manages (curator_id, gallery_id, starting_date)
VALUES(5, 5, "2020-01-01");

INSERT INTO Oversees (curator_id, exhibit_id)
VALUES(2, 1);

INSERT INTO Oversees (curator_id, exhibit_id)
VALUES(1, 2);

INSERT INTO Oversees (curator_id, exhibit_id)
VALUES(6, 3);

INSERT INTO Oversees (curator_id, exhibit_id)
VALUES(4, 4);

INSERT INTO Oversees (curator_id, exhibit_id)
VALUES(7, 5);

INSERT INTO Transfers (art_id, origin, date, destination)
VALUES(1, 1, "2021-09-10", 2);

INSERT INTO Transfers (art_id, origin, date, destination)
VALUES(2, 2, "1924-08-13", 3);

INSERT INTO Transfers (art_id, origin, date, destination)
VALUES(4, 4, "1966-07-16", 1);

INSERT INTO Transfers (art_id, origin, date, destination)
VALUES(3, 3, "1999-06-17", 2);

INSERT INTO Transfers (art_id, origin, date, destination)
VALUES(6, 1, "2001-05-03", 2);

INSERT INTO Transfers (art_id, origin, date, destination)
VALUES(5, 2, "2022-04-02", 1);

INSERT INTO Transfers (art_id, origin, date, destination)
VALUES(4, 3, "2023-03-19", 4);

INSERT INTO Transfers (art_id, origin, date, destination)
VALUES(11, 5, "1852-02-23", 2);

INSERT INTO Transfers (art_id, origin, date, destination)
VALUES(12, 5, "2024-01-27", 3);

INSERT INTO Visits (customer_id, exhibit_id, time, price)
VALUES (1, 1, "2025-04-02 23:59:59.10", 50);

INSERT INTO Visits (customer_id, exhibit_id, time, price)
VALUES (2, 2, "2026-11-31 23:59:59.10", 50);

INSERT INTO Visits (customer_id, exhibit_id, time, price)
VALUES (3, 3, "2025-07-21 23:59:59.10", 40);

INSERT INTO Visits (customer_id, exhibit_id, time, price)
VALUES (4, 4, "2027-03-15 23:59:59.10", 25);

INSERT INTO Visits (customer_id, exhibit_id, time, price)
VALUES (5, 5, "2026-08-03 23:59:59.10", 50);

INSERT INTO CustomerVisitsExhibit (customer_id, time, exhibit_id)
VALUES(1, "2025-04-02 23:59:59.10", 1);

INSERT INTO CustomerVisitsExhibit (customer_id, time, exhibit_id)
VALUES(2, "2026-11-31 23:59:59.10", 2);

INSERT INTO CustomerVisitsExhibit (customer_id, time, exhibit_id)
VALUES(3, "2025-07-21 23:59:59.10", 3);

INSERT INTO CustomerVisitsExhibit (customer_id, time, exhibit_id)
VALUES(4, "2027-03-15 23:59:59.10", 4);

INSERT INTO CustomerVisitsExhibit (customer_id, time, exhibit_id)
VALUES(5, "2026-08-03 23:59:59.10", 5);

INSERT INTO ExhibitCharges (exhibit_id, price, price_group)
VALUES(1, 50, "Adult");

INSERT INTO ExhibitCharges (exhibit_id, price, price_group)
VALUES(1, 40, "Student");

INSERT INTO ExhibitCharges (exhibit_id, price, price_group)
VALUES(1, 25, "Senior");

INSERT INTO ExhibitCharges (exhibit_id, price, price_group)
VALUES(2, 50, "Adult");

INSERT INTO ExhibitCharges (exhibit_id, price, price_group)
VALUES(2, 40, "Student");

INSERT INTO ExhibitCharges (exhibit_id, price, price_group)
VALUES(2, 25, "Senior");

INSERT INTO ExhibitCharges (exhibit_id, price, price_group)
VALUES(3, 50, "Adult");

INSERT INTO ExhibitCharges (exhibit_id, price, price_group)
VALUES(3, 40, "Student");

INSERT INTO ExhibitCharges (exhibit_id, price, price_group)
VALUES(3, 25, "Senior");

INSERT INTO ExhibitCharges (exhibit_id, price, price_group)
VALUES(4, 50, "Adult");

INSERT INTO ExhibitCharges (exhibit_id, price, price_group)
VALUES(4, 40, "Student");

INSERT INTO ExhibitCharges (exhibit_id, price, price_group)
VALUES(4, 25, "Senior");

INSERT INTO ExhibitCharges (exhibit_id, price, price_group)
VALUES(5, 50, "Adult");

INSERT INTO ExhibitCharges (exhibit_id, price, price_group)
VALUES(5, 40, "Student");

INSERT INTO ExhibitCharges (exhibit_id, price, price_group)
VALUES(5, 25, "Senior");