

SOEN 363: Data Systems for Software Engineers

Assignment 1

Nathan Grenier

October 10, 2024

Fall 2024

Q1. [40 Points] Consider the following system description for a museum information systems:

A museum has multiple departments. A museum has collections. A collection belongs to a department.

Art-objects belong to a collection. An art-object has a unique acquisition number, object type, title, description, dimension, production date, and production country. Examples of object types are paintings, sculptures, etc. An art-object may be linked to an artist. Some art-objects are just without one. An art-object belongs to a period (i.e. Roman Empire).

An exhibition takes place in a museum which exhibits various art objects from various collections. An exhibition has a start and end date, as well as a home department. Art objects may be borrowed temporarily from various museums for a specific exhibition.

A) [20 pts] Design a data model that implements the above description. Identify the entities, relations, attributes, keys, and their cardinalities. Identify strong and weak entities and relations, if applicable. Represent the end result in the form of an ERD diagram. single arrow - no arrow notation for cardinalities and single line - double line for partial and total participation [??].

Note: The software I used couldn't do dashed underlined text. Therefore, bold + underline + italics represents a partial key.

3

```

11     PRIMARY KEY (EXHI_ID, MUS_ID),
12     FOREIGN KEY (MUS_ID) REFERENCES Museum(MUS_ID)
13     FOREIGN KEY (DEP_HOME_ID) REFERENCES Departments(DEP_ID)
14 );
15
16 CREATE TABLE IF NOT EXISTS Departments (
17     DEP_ID INT,
18     MUS_ID INT,
19     PRIMARY KEY (DEP_ID, MUS_ID),
20     FOREIGN KEY (MUS_ID) REFERENCES Museum(MUS_ID)
21 );
22
23 CREATE TABLE IF NOT EXISTS Collections (
24     COLL_ID INT,
25     DEP_ID INT,
26     EXHI_ID INT,
27     PRIMARY KEY (COLL_ID, DEP_ID),
28     FOREIGN KEY (DEP_ID) REFERENCES Departments(DEP_ID),
29     FOREIGN KEY (EXHI_ID) REFERENCES Exhibitions(EXHI_ID)
30 );
31
32 CREATE TABLE IF NOT EXISTS Countries (
33     COUNTRY_ID INT PRIMARY KEY,
34     COUNTRY_NAME VARCHAR(40)
35 );
36
37 CREATE TABLE IF NOT EXISTS Artists (
38     ARTIST_ID INT PRIMARY KEY,
39     ARTIST_NAME VARCHAR(40)
40 );
41

```

```

42 CREATE TABLE IF NOT EXISTS ArtObjectTypes (
43     ARTOBJ_TYPE_ID INT PRIMARY KEY,
44     ARTOBJ_TYPE_NAME VARCHAR(40)
45 );
46
47 CREATE TABLE IF NOT EXISTS Periods (
48     PER_ID INT PRIMARY KEY,
49     PER_NAME VARCHAR(40) NOT NULL,
50     PER_START_DATE DATE NOT NULL,
51     PER_END_DATE DATE NOT NULL
52 );
53
54 CREATE TABLE IF NOT EXISTS Dimensions (
55     DIM_ID INT,
56     ARTOBJ_ID INT,
57     DIM_LENGTH FLOAT,
58     DIM_WIDTH FLOAT,
59     DIM_DEPTH FLOAT,
60     PRIMARY KEY (DIM_ID, ARTOBJ_ID)
61 );
62
63 CREATE TABLE IF NOT EXISTS ArtObjects (
64     ARTOBJ_ACQUISITION_NUM INT PRIMARY KEY,
65     ARTOBJ_TYPE_ID INT NOT NULL,
66     COLL_ID INT NOT NULL,
67     ARTIST_ID INT,
68     COUNTRY_ID INT,
69     PER_ID INT NOT NULL,
70     DIM_ID INT NOT NULL,
71     ARTOBJ_TITLE VARCHAR(40),
72     ARTOBJ_DESC VARCHAR(100),

```

```

73     ARTOBJ_PROD_DATE DATE,
74     FOREIGN KEY (ARTOBJ_TYPE_ID) REFERENCES
      ⇨ ArtObjectTypes(ARTOBJ_TYPE_ID),
75     FOREIGN KEY (PER_ID) REFERENCES Periods(PER_ID),
76     FOREIGN KEY (DIM_ID) REFERENCES Dimensions(DIM_ID),
77     FOREIGN KEY (COLL_ID) REFERENCES Collections(COLL_ID),
78     FOREIGN KEY (ARTIST_ID) REFERENCES Artists(ARTIST_ID),
79     FOREIGN KEY (COUNTRY_ID) REFERENCES Countries(COUNTRY_ID)
80 );
81
82 CREATE TABLE IF NOT EXISTS Borrowed (
83     EXHI_ID INT,
84     ARTOBJ_ACQUISITION_NUM INT,
85     B_DATE DATE,
86     B_RETURN_DATE DATE,
87     PRIMARY KEY (EXHI_ID, ARTOBJ_ACQUISITION_NUM),
88     FOREIGN KEY (EXHI_ID) REFERENCES Exhibitions(EXHI_ID),
89     FOREIGN KEY (ARTOBJ_ACQUISITION_NUM) REFERENCES
      ⇨ ArtObjects(ARTOBJ_ACQUISITION_NUM)
90 );

```

- C) [10 pts] Provide a database instance that demonstrate an exhibition with local (from home department) and borrowed art objects.

```
1  INSERT INTO Museum
2      (MUSEUM_ID)
3  VALUES
4      (1);
5
6  INSERT INTO Departments
7      (DEP_ID, MUS_ID)
8  VALUES
9      (12, 1),
10     (22, 1),
11     (55, 1);
12
13 INSERT INTO Exhibitions
14     (EXHI_ID, MUS_ID, DEP_HOME_ID, EXHI_START_DATE, EXHI_END_DATE)
15 VALUES
16     (123, 1, 12, '2023-01-01', '2023-01-31' ),
17     (321, 1, 22, '2024-02-01', '2024-07-01' );
18
19 INSERT INTO Collections
20     (COLL_ID, DEP_ID, EXHI_ID)
21 VALUES
22     (1, 22, 123),
23     (2, 12, 123),
24     (3, 55, 321);
25
26 INSERT INTO Countries
27     (COUNTRY_ID, COUNTRY_NAME)
28 VALUES
29     (1, 'Canada'),
```

```

30     (2, 'USA'),
31     (3, 'Mexico'),
32     (4, 'Italy');
33
34 INSERT INTO Artists
35     (ARTIST_ID, ARTIST_NAME)
36 VALUES
37     (1, 'Pablo Picasso'),
38     (2, 'Vincent Van Gogh'),
39     (3, 'Leonardo Da Vinci'),
40     (4, 'Myron');
41
42 INSERT INTO ArtObjectTypes
43     (ARTOBJ_TYPE_ID, ARTOBJ_TYPE_NAME)
44 VALUES
45     (1, 'Painting'),
46     (2, 'Sculpture'),
47     (3, 'Photography');
48
49 INSERT INTO Periods
50     (PER_ID, PER_NAME, PER_START_DATE, PER_END_DATE)
51 VALUES
52     (1, 'Renaissance', '1400-01-01', '1600-01-01'),
53     (2, 'Baroque', '1600-01-01', '1700-01-01'),
54     (3, 'Impressionism', '1800-01-01', '1900-01-01');
55
56 INSERT INTO ArtObjects
57     (ARTOBJ_ACQUISITION_NUM, ARTOBJ_TYPE_ID, COLL_ID, ARTIST_ID,
58     ↪ COUNTRY_ID, PER_ID, DIM_ID, ARTOBJ_TITLE, ARTOBJ_DESC,
59     ↪ ARTOBJ_PROD_DATE)
60 VALUES

```



```

59      (1, 1, 3, 1, 1, 1, 1, 'Starry Night', 'A painting of a starry night',
        ↪ '1889-06-01'),
60      (2, 1, 2, 2, 2, 2, 2, 'Mona Lisa', 'A painting of Mona Lisa',
        ↪ '1503-01-01'),
61      (3, 3, 1, 3, 3, 3, 3, 'The Persistence of Memory', 'A painting of
        ↪ melting clocks', '1931-01-01'),
62      (4, 2, 3, 4, 4, 3, 3, 'Discobolus', 'A man throwing a disc',
        ↪ '1000-01-01');
63
64  INSERT INTO Dimensions
65      (DIM_ID, ARTOBJ_ID, DIM_LENGTH, DIM_WIDTH, DIM_DEPTH)
66  VALUES
67      (1, 1, 10.0, 10.0, 10.0),
68      (2, 2, 20.0, 20.0, 20.0),
69      (3, 3, 30.0, 30.0, 30.0);
70
71  INSERT INTO Borrowed
72      (EXHI_ID, ARTOBJ_ACQUISITION_NUM, B_DATE, B_RETURN_DATE)
73  VALUES
74      (123, 1, '2023-01-01', '2023-01-31');

```

Q2. [40 Points] Consider the following system description to manage group or private lessons:

Consider an organization that offers group or private lessons (of various types, such as yoga, swimming, etc.) to clients. The organization owns or rents space (gyms, rooms, swimming pools) in various locations in possibly different cities across the province and each location is made available over some given schedule, where a schedule is essentially a sequence of day-time slots. E.g. “The EV-Building gym Room 7, in Montreal, is available for Judo classes on Sundays from 12 noon to 3PM, from September 1st to November 30th, 2024.” The length of a time slot is not fixed (and you are free to create schedules as you see fit). For example, a swimming lesson would normally be for half hour, but a Judo class would normally be for one hour.

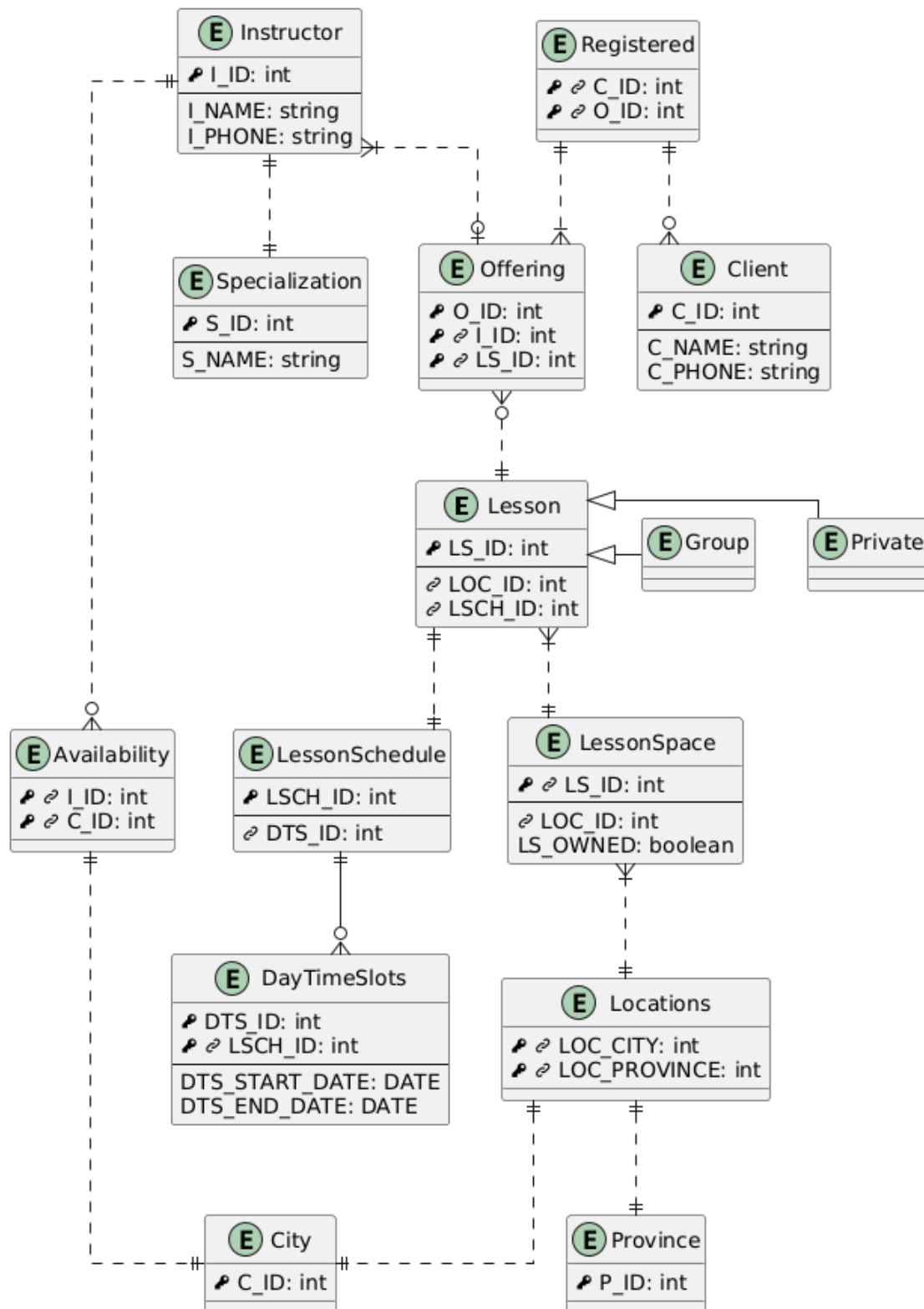
A given location can only accommodate one lesson at any given time slot on a given day. The same type of lesson can be offered in both modes (private, group). For example, at the same location the organization would offer private swimming lessons and group swimming lessons.

The organization does not have permanent instructors, but it hires seasonal instructors of various specializations. An instructor would register with the system by entering their credentials (name and phone number) and specialization as well as to register their availability in one or possibly several cities. For example, “Grace (514 - ...) is a swim instructor, available to work in Montreal and Laval.”

Once registered, an instructor can subsequently take on possibly several offerings that the organization makes available. All offerings are made available to potential instructors, but only those that have been taken by instructors are made available to the public in order to attract clients. For example, “We offer private and group Judo classes in EV-Building on Sundays from 12PM to 3PM from 1.Sep to 30.Nov as follows: 12:00 - 13:00. Group. Instructor: ... 13:00 - 14:00. Group. ... 14:00 - 14:30. Private. ... 14:30 - 15:00. Private. ... We offer swimming classes at ”

A) [25 pts] Design a data model that captures the above description. Identify the entities, relations, attributes, keys, and their cardinalities. Identify strong and

weak entities and relations, if applicable. Represent the end result in the form of an ERD diagram, using chen notation [??].



B) [6 pts] Provide the schema and explicitly identify the domains of each attribute.

```
1 CREATE TABLE City (  
2     C_ID INT PRIMARY KEY  
3 );  
4  
5 CREATE TABLE Province (  
6     P_ID INT PRIMARY KEY  
7 );  
8  
9 CREATE TABLE Locations (  
10     LOC_CITY INT,  
11     LOC_PROVINCE INT,  
12     PRIMARY KEY (LOC_CITY, LOC_PROVINCE),  
13     FOREIGN KEY (LOC_CITY) REFERENCES City(C_ID),  
14     FOREIGN KEY (LOC_PROVINCE) REFERENCES Province(P_ID)  
15 );  
16  
17 CREATE TABLE Lesson (  
18     LS_ID INT PRIMARY KEY,  
19     LOC_ID INT,  
20     LSCH_ID INT,  
21     FOREIGN KEY (LOC_ID) REFERENCES Locations(LOC_CITY),  
22     FOREIGN KEY (LSCH_ID) REFERENCES LessonSchedule(LSCH_ID)  
23 );  
24  
25 CREATE TABLE Private (  
26     LS_ID INT PRIMARY KEY,  
27     FOREIGN KEY (LS_ID) REFERENCES Lesson(LS_ID)  
28 );  
29  
30 CREATE TABLE Group (  
31     LS_ID INT PRIMARY KEY,  
32     FOREIGN KEY (LS_ID) REFERENCES Lesson(LS_ID)
```

```

31     LS_ID INT PRIMARY KEY,
32     FOREIGN KEY (LS_ID) REFERENCES Lesson(LS_ID)
33 );
34
35 CREATE TABLE LessonSpace (
36     LS_ID INT,
37     LOC_ID INT,
38     LS_OWNED BOOLEAN,
39     PRIMARY KEY (LS_ID),
40     FOREIGN KEY (LS_ID) REFERENCES Lesson(LS_ID),
41     FOREIGN KEY (LOC_ID) REFERENCES Locations(LOC_CITY)
42 );
43
44 CREATE TABLE LessonSchedule (
45     LSCH_ID INT PRIMARY KEY,
46     DTS_ID INT,
47     FOREIGN KEY (DTS_ID) REFERENCES DayTimeSlots(DTS_ID)
48 );
49
50 CREATE TABLE DayTimeSlots (
51     DTS_ID INT PRIMARY KEY,
52     LSCH_ID INT,
53     DTS_START_DATE DATE,
54     DTS_END_DATE DATE,
55     FOREIGN KEY (LSCH_ID) REFERENCES LessonSchedule(LSCH_ID)
56 );
57
58 CREATE TABLE Instructor (
59     I_ID INT PRIMARY KEY,
60     I_NAME VARCHAR(255),
61     I_PHONE VARCHAR(255)

```

```

62 );
63
64 CREATE TABLE Specialization (
65     S_ID INT PRIMARY KEY,
66     S_NAME VARCHAR(255)
67 );
68
69 CREATE TABLE Offering (
70     O_ID INT,
71     I_ID INT,
72     LS_ID INT,
73     PRIMARY KEY (O_ID, I_ID, LS_ID),
74     FOREIGN KEY (I_ID) REFERENCES Instructor(I_ID),
75     FOREIGN KEY (LS_ID) REFERENCES Lesson(LS_ID)
76 );
77
78 CREATE TABLE Availability (
79     I_ID INT,
80     C_ID INT,
81     PRIMARY KEY (I_ID, C_ID),
82     FOREIGN KEY (I_ID) REFERENCES Instructor(I_ID),
83     FOREIGN KEY (C_ID) REFERENCES City(C_ID)
84 );
85
86 CREATE TABLE Client (
87     C_ID INT PRIMARY KEY,
88     C_NAME VARCHAR(255),
89     C_PHONE VARCHAR(255)
90 );
91
92 CREATE TABLE Registered (

```

```
93     C_ID INT,  
94     O_ID INT,  
95     PRIMARY KEY (C_ID, O_ID),  
96     FOREIGN KEY (C_ID) REFERENCES Client(C_ID),  
97     FOREIGN KEY (O_ID) REFERENCES Offering(O_ID)  
98 );
```

- C) [9 pts] Provide a database instance that demonstrate offers to a sample instructor, one of which is accepted, one is rejected, and one is pending response. Demonstrate a group and a private lesson in your example.

```
1  INSERT INTO City (C_ID) VALUES (1), (2);
2  INSERT INTO Province (P_ID) VALUES (1), (2);
3  INSERT INTO Locations (LOC_CITY, LOC_PROVINCE) VALUES (1, 1), (2, 2);
4
5  INSERT INTO LessonSchedule (LSCH_ID, DTS_ID) VALUES (1, 1), (2, 2);
6  INSERT INTO DayTimeSlots (DTS_ID, LSCH_ID, DTS_START_DATE,
   ↪ DTS_END_DATE) VALUES (1, 1, '2023-01-01', '2023-01-31'), (2, 2,
   ↪ '2023-02-01', '2023-02-28');
7
8  INSERT INTO Lesson (LS_ID, LOC_ID, LSCH_ID) VALUES (1, 1, 1), (2, 2,
   ↪ 2);
9  INSERT INTO Private (LS_ID) VALUES (1);
10 INSERT INTO Group (LS_ID) VALUES (2);
11
12 INSERT INTO LessonSpace (LS_ID, LOC_ID, LS_OWNED) VALUES (1, 1, TRUE),
   ↪ (2, 2, FALSE);
13
14 INSERT INTO Instructor (I_ID, I_NAME, I_PHONE) VALUES (1, 'John Doe',
   ↪ '123-456-7890');
15
16 INSERT INTO Specialization (S_ID, S_NAME) VALUES (1, 'Math'), (2,
   ↪ 'Science');
17
18 INSERT INTO Offering (O_ID, I_ID, LS_ID) VALUES (1, 1, 1), (2, 1, 2),
   ↪ (3, 1, 1);
19
20 INSERT INTO Client (C_ID, C_NAME, C_PHONE) VALUES (1, 'Alice',
   ↪ '987-654-3210'), (2, 'Bob', '654-321-0987');
```



```
21
22 INSERT INTO Registered (C_ID, O_ID) VALUES (1, 1), (2, 2);
23
24 -- Sample data for offers
25 -- Offer 1: Accepted
26 INSERT INTO Offering (O_ID, I_ID, LS_ID) VALUES (1, 1, 1);
27 -- Offer 2: Rejected
28 INSERT INTO Offering (O_ID, I_ID, LS_ID) VALUES (2, 1, 2);
29 -- Offer 3: Pending
30 INSERT INTO Offering (O_ID, I_ID, LS_ID) VALUES (3, 1, 1);
```

Q3. [20 Points] Represent the ERD of either questions in crow's foot notation using PlantUml [??], [??]. Make sure you submit both PlantUML source and the diagram image.

Question 1 ERD (Crow's Foot Notation):

