



Statistical Methods for Database Integration

Examination

DATABASES

The exam consists in two parts:

- 1) PART A: **The exam is closed-book, closed-notes;**
- 2) PART B: **You are allowed to use lecture and labs notes.**

Each questions is assigned points expressed in cents.

PART A

Ex. 1

- (a) **(10 points)** “Currently the Web shows a variety of data in effective way”. In order to analyse cinemas market consider the following worldwide box office data, in the current year, and describe it by means of the XML language. Describe a very **small** part of the data, paying attention to **units**, and data that could be **grouped** or **omitted**.

2022 ▾						
Rank ^	Release Group	Worldwide ↕	Domestic ↕	% ↕	Foreign ↕	% ↕
1	Top Gun: Maverick	\$1,453,450,344	\$705,650,344	48.6%	\$747,800,000	51.4%
2	Jurassic World Dominion	\$994,975,130	\$375,859,130	37.8%	\$619,116,000	62.2%
3	Doctor Strange in the Multiverse of Madness	\$955,775,804	\$411,331,607	43%	\$544,444,197	57%

Figure 1: source: <https://www.boxofficemojo.com/year/world/>



[Sol.:]

```

▼<box-office year="2022">
  ▼<ranking>
    <rank> 2 </rank>
    <movie> Jurassic World Dominion </movie>
    ▼<amount unit="$">
      <domestic> 375859130 </domestic>
      <foreign> 619116000 </foreign>
    </amount>
  </ranking>
</box-office>

```

Figure 2: XML file

- (b) **(10 points)** A table **catalogue** should be created in order to show movies data on the web. We have to register **mandatory**: title, year, length, director, producer, box office, awarded (yes/no).

Write the SQL statement to create the table, paying attention to data type and to the primary key.

Then write two **INSERT** statement.

[Sol.:]

```

CREATE TABLE Catalogue(
  title VARCHAR(50),
  year CHAR(4),
  length INT,
  director VARCHAR(30),
  producer VARCHAR(30),
  boxOffice NUMERIC(12,0),
  awarded BOOLEAN,
  primary key(title, year)
);

INSERT INTO Catalogue VALUE('Top Gun: Maverick', '2022', 131,
                             'Joseph Kosinski', 'Paramount', 1453450344, FALSE);
INSERT INTO Catalogue VALUE('Jurassic World Dominion', '2022', 146,
                             'Colin Trevorrow', 'Universal', 994975130, FALSE);

```

- (c) **(Optional: 5 points)** Data integration really represents a challenge in data management. ODBC technology can be easily used to integrate some applications and relational databases. What does ODBC technology enable? Describe the experience of data integration we have set up in the lab.

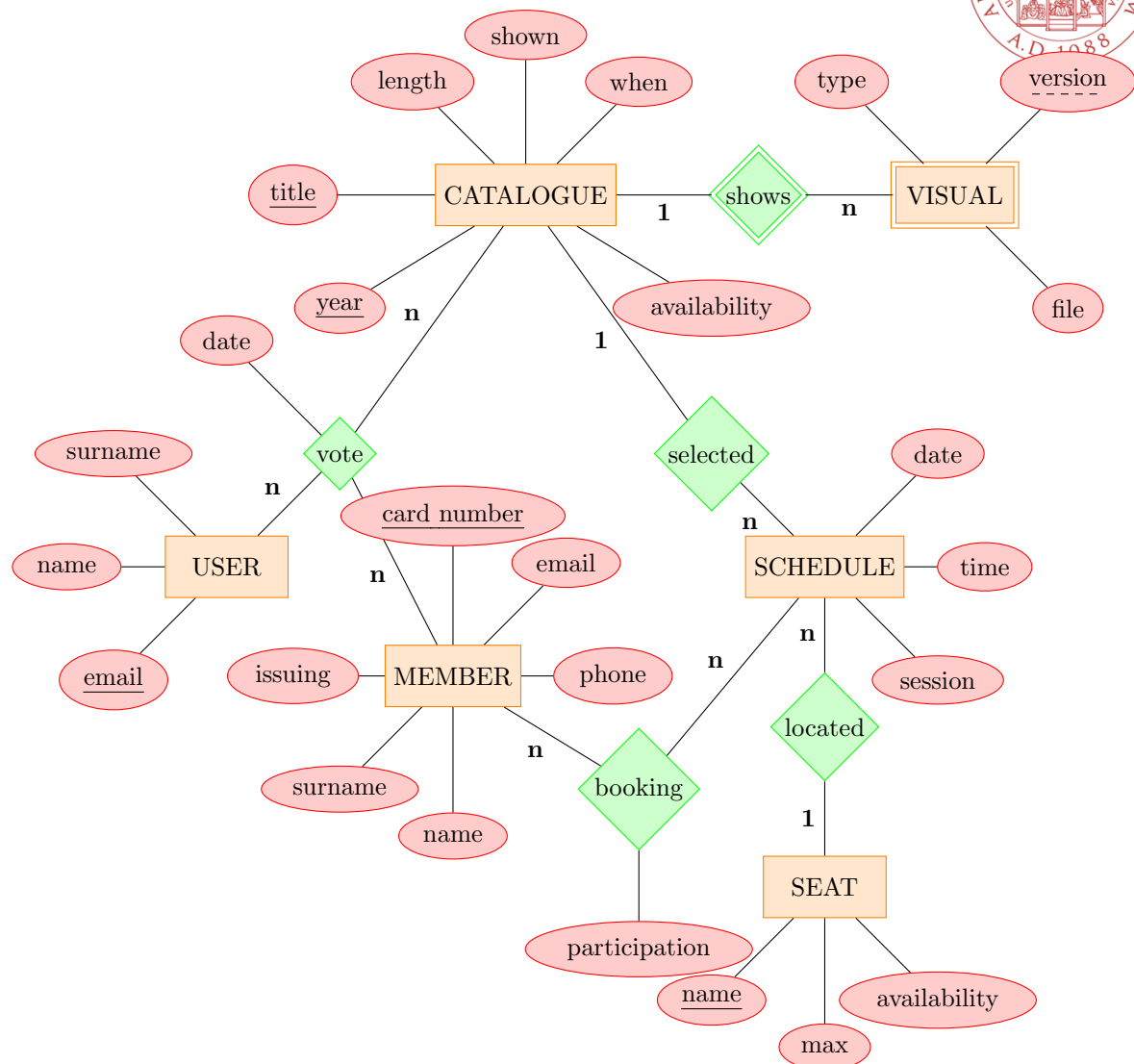
[Sol.: See teaching material]

**Es. 2 - Data Modeling**

- (1) **(35 points)** “Apps for smartphone are valid and friendly tools to offer cultural services”. The municipality offers the great opportunity of free “cineforum”, where citizen can choose movies they mostly like. **App** that should accept movies preferences, member registration and participation is considered a valid solution. App should be supported by a database.

Draw the E/R diagram that capture the requirements stated below. Use “ID” as key only if strictly necessary.

- (a) Everyone has the opportunity to vote one or more movies registered into a **catalogue** he/she likes the most. It represents a collection of movies, each one described by a title, year of realization, length, if it is available to show (yes/no), if it has already shown and when.
- (b) A movie can have one or more media visuals: poster, trailer, ... a **visual** has a type, a file, and could have many versions. A visual is strictly related to the movie, hence it should be designed has weak entity.
- (c) The app allows everyone, to vote his/her best movie in the catalogue - which can be voted just once - , it is sufficient to be a **user** of the App. To install the app you have to specify your name, surname and the email.
- (d) In order to book a seat at the cinema you must be **member**. The following data shall be registered: card number, name, surname, email and phone number, issuing date. **A member can continue to vote his/her preferite movies.**
- (e) Movies can be shown in various type of **seats**: park (open air movies), theatre, cinema, school, library, Each seat is identified by a name, report if it is currently available or temporarily not available, maximum number of seats.
- (f) Every six months, the most voted movies are scheduled for the next weeks, considering if they are recently shown or not. **Schedule** identifies a movie to be shown in a seat at a specified date and time. The movie can be a part of a special session, in this case report the title of the session.
- (g) Members - not general users - can book a seat for one or more movie shows (it is a very simple relationship). A member can go to the seat just showing a personal QRCode. The participation is registered (yes/no).



- (2) (Optional: 5 points). Write the SQL statement to CREATE the “relations” that describes the **catalogue** and **visual**.

```
CREATE TABLE Catalogue(
  title VARCHAR(50),
  year CHAR(4),
  length INT,
  shown BOOLEAN,
  date DATE,
  availability BOOLEAN,
  primary key(title, year)
);
```

```
CREATE TABLE Visual(
  title VARCHAR(50),
  year CHAR(4),
  version INT,
  type VARCHAR(30),
  file BLOB,
  FOREIGN KEY (title, year)
    REFERENCES Catalogue(title, year)
);
```



PARTE B

Es. 3 - SQL (45 points) Let assume the database “online-market”.

- (1) Region(name, description)
- (2) Sheet(ID, description, Region.name)
- (3) Producer(name, description)
- (4) Produced(Producer.name, Sheet.ID)
- (5) Ingredient(name, description)
- (6) Made(Ingredient.name, Sheet.ID)
- (7) Menu(name, description, main)
- (8) Food(name, unit, weight, label, price, startDate, endDate, Menu.main_name, Sheet.ID)
- (9) GiftBasket(name, description)
- (10) BasketCombines(GiftBasket.name, Food.name, Food.unit, Food.weight)
- (11) User(ID, date, time, network_info)
- (12) Consulted(User.ID, Food.name, Food.unit, Food.weight, time)
- (13) Selected(User.ID, Food.name, Food.unit, Food.weight, time, quantity)

Questions

- 1) The marketing would like to analyze the part of the day users prefer to visit the online market, especially during working hours. Report for each hour of the day how many users have been connected at that hour [*tip*: use the function `hour()` to extract the hour from the complete time]. Consider only working hours, that is from 9 (included) to 13 and from 14 (included) to 18.

[*Sol.*]

```
SELECT hour(time) AS slot, COUNT(*)
  FROM user
 WHERE (hour(time) >= 9 AND hour(time) < 13)
      OR (hour(time) >= 14 AND hour(time) < 18)
 GROUP BY hour(time);
```



- 2) We have the suspicion that a competitor accesses very frequently to the web store, just to consult - and maybe copy - food offers. List all food consultations for which there exists at least one access (table user) in the month of June coming from a network starting with **112** or **113**.

[Sol.]

```
SELECT *
  FROM consulted
 WHERE EXISTS (SELECT *
                FROM user
               WHERE month(date) = 6
                     AND (network_info LIKE '112%' OR network_info LIKE '113')
                     AND ID = consulted.ID);
```

- 3) Marketing would like to know the entity value of each basket [we consider as basket which food products a user has selected. The value is simply the quantity of each item selected multiplied by the corresponding price]. Therefore compute and report for each user ID the computed value. Consider only basket values greater than or equal to 50 €.

[Sol.]

```
SELECT ID, SUM(quantity*price) AS total
  FROM food, selected
 WHERE food.name = selected.food_name
       AND food.weight = selected.food_weight
       AND food.unit = selected.food_unit
 GROUP BY ID
 HAVING total >= 50;
```