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 offers a variety of data, even if it is embodied into HTML code". Marketing needs to compare Internet service offers of different providers. Consider the following Internet offers data and describe it by means of the XML language (**consider just one offer at your choice!**)

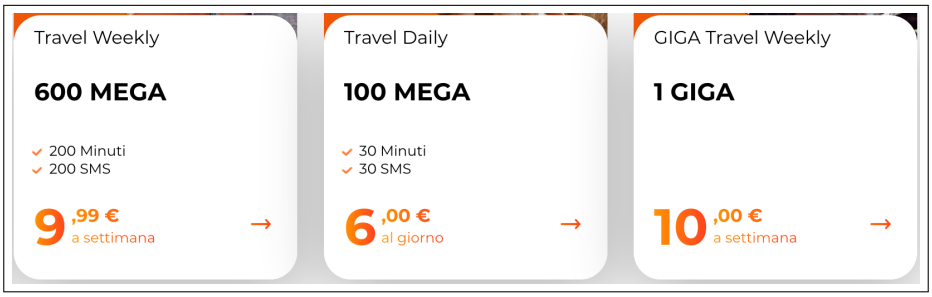


Figure 1: source: **www.windtre.it**: Internet offers to travel abroad

[*Sol.*:]

```
▼<Internet-offers>
  ▼<offer>
     <name> Travel Weekly </name>
    ▼<options>
       <internet unit="mega"> 600 </internet>
       <minutes> 200 </minutes>
       <SMS> 200 </SMS>
     </options>
     <price frequency="weekly" unit="€"> 9,99 </price>
   </offer>
 </Internet-offers>
```

Figure 2: **XML file**

(b) **(10 points)** Since 2010, offers for voice and sms services have been collected and stored in the table below. Consider to store also the data of the picture above. It looks that the table shall be altered in order to store it. Modify properly the SQL statements avoiding to loose data/information already registered. Further since now a provider is identified by its name and its country modify `PRIMARY KEY` declaration.

```
CREATE TABLE offer(
   provider VARCHAR(10) PRIMARY KEY,
   country CHAR(2),
   offer_name VARCHAR(10),
   minutes NUMERIC(6,2),
   sms NUMERIC(5,2),
   price NUMERIC(4,1)
);
```

```
CREATE TABLE offer(
   provider VARCHAR(10),
   country CHAR(2),
   offer_name VARCHAR(30),
   minutes NUMERIC(6,2),
   sms NUMERIC(6,2),
   internet NUMERIC(4,0),
   int_unit CHAR(4),
   price NUMERIC(4,2),
   frequency VARCHAR(20),
   PRIMARY KEY (provider, country)
);
```

(c) **(Optional: 5 points) Scraping** is the process to implement to get data from Web pages. Describe shortly this process independently of the used programming language.

[*Sol.*: See teaching material]

Es. 2 - Data Modeling

(1) **(35 points)** "Just one click and you can locate and book the best cakes of the tradition, you identify your best favorite cake (panettone, torrone, pastiera, ...), locate one bakery close to you, check the availability, book the cake and finally get and taste it". The goal is to design a database to support the 'Web application' which allows user to "locate and book " the best favorite traditional cake.

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

(a) **Traditional cakes** are stored with a common name, an additional name, the region of reference, year of the first preparation (when it is available), a picture.

(b) Typically, many **"variants"** are available. Along time bakers have been created "modern" cakes, maybe just adding or replacing ingredients. Therefore these "variants" should be recorded with the additional name.

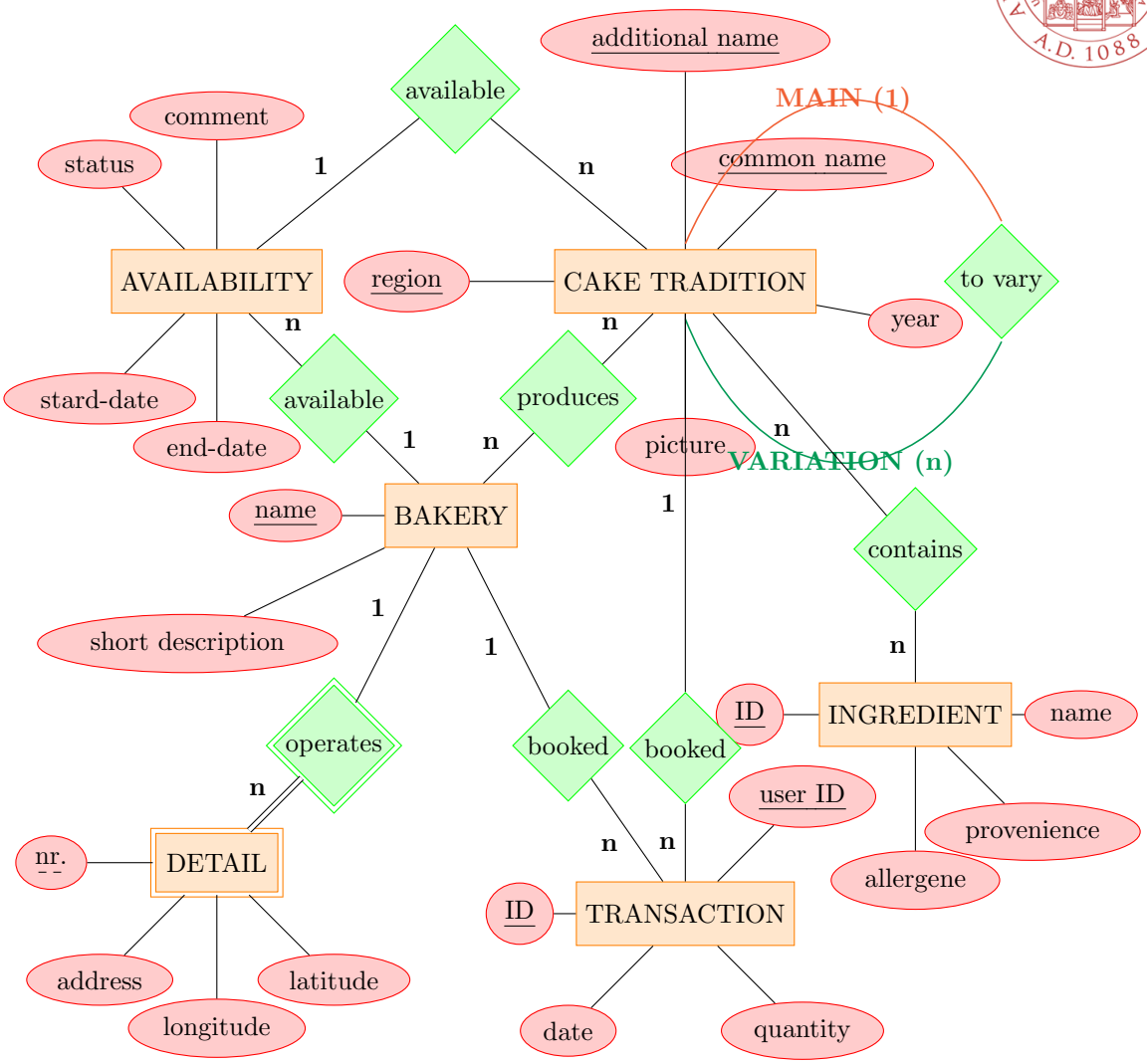
(c) For each cakes, and its variants, should be displayed the main **ingredients**, so the name, provenience, and if it could contains allergene (Yes or No).

(d) **Bakeries** which offer cakes of the tradition are registered, the name and a short description.

(e) A bakery could operate on the region with different shops, therefore the following **details** should be registered aiming to show to the user the best close shop, that is where to take the booked cake. Specifically details are related to the address, geographical coordinates (longitude, latitude) in order to compute the distance.

(f) A bakery generally is not able to offer all cakes along the year, since for instance cakes are related to seasonality (panettone, torrone) or seasonal ingredients. Availability of cakes offered by each bakery must be registered in order to allow and accept bookings. **Availability** is described by a period: start date and end date, a comment, a status (True of False whenever this availability is confirmed or not).

(g) Whenever a user locate a bakery, for its best favorite cakes and the user decides to book it a **transaction** is registered.



(2) **(Optional: 5 points)**. Write the SQL statement to `CREATE` the "relation" that describes all **availability** items.

```
CREATE TABLE availability(
   bakery VARCHAR(30),
   start_date DATE,
   end_date DATE,
   status VARCHAR(5),
   comment VARCHAR(30),
   FOREIGN KEY (bakery) REFERENCES backery(name)
);
```

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 reconsider number and value of food products to offer as speciality of a region. The request is to extract the region name for which the online market offers the largest number of products, moreover for that region the cheapest and the expensive products offered. [**Tip:** the query return just one region!] **Use the explicit JOIN if it is necessary to join tables.**

   [*Sol.*]

```
SELECT region_name, MIN(food.price), MAX(food.price)
  FROM sheet, food
 WHERE food.sheet_ID = sheet.ID
GROUP BY region_name
HAVING COUNT(ID) >= ALL(SELECT COUNT(ID)
                          FROM sheet
                        GROUP BY region_name);
```

2) At this time (so considering the whole data) are there products of type 'cream', which nobody has neither consulted nor selected? [**Tip**: check the existence or not existence!]

[*Sol.*]

```
SELECT name, unit, weight
FROM food
WHERE food.menu_name LIKE '%cream%'
AND NOT EXISTS (SELECT *
                  FROM selected S
                 WHERE S.food_name = food.name
                   AND S.food_unit = food.unit
                   AND S.food_weight = food.weight)
AND NOT EXISTS (SELECT *
                  FROM consulted C
                 WHERE C.food_name = food.name
                   AND C.food_unit = food.unit
                   AND C.food_weight = food.weight);
```

3) Report food products which has been consulted and selected (both). Display food products, the quantity selected, the date and time of consulted and selection. **Consider only the case where date and time of consulted are previous of date and time of selection.**

[*Sol.*]

```
SELECT C.food_name, C.food_unit, C.food_weight, S.quantity,
                          UC.date, US.date, C.time, S.time
  FROM consulted C, selected S, user UC, user US
 WHERE C.food_name = S.food_name
   AND C.food_unit = S.food_unit
   AND C.food_weight = S.food_weight
   AND C.ID = UC.ID
   AND S.ID = US.ID
   AND C.time < S.time,
   AND UC.date < US.date;
```