

# SIDBMS - Simulation 3. 01 Dec 2020

prof. Andrea Pinna

## Exercise 1.

Write a program that reads the data from the file "stazione\_bike\_sharing.csv". The program extracts all the values of the ID\_BIKE\_SHARING fields and STAZIONE writes them to a json format file.

Example:

Input file:

```
WKT_GEOM;ID_BIKE_SHARING;STAZIONE;URL;COORD_X;COORD_Y
```

```
POINT (1395368.54934478  
4992449.00938128);1;Paravia;http://www.tobike.it/frmLeStazioni.aspx;1395368,  
54934478;4992449,00938128
```

```
POINT (1395197.62755516 4991980.32337951);2;Porta Susa  
2;http://www.tobike.it/frmLeStazioni.aspx;1395197,62755516;4991980,32337951
```

Output file:

```
[  
  {  
    "ID_BIKE_SHARING": "1",  
    "STAZIONE": "Paravia"  
  },  
  {  
    "ID_BIKE_SHARING": "2",  
    "STAZIONE": "Porta Susa 2"  
  }  
]
```

## Exercise 2

1. Write the instructions to connect to the DBMS and if it does not exist, create the "DBbikesharing" database and use it. Then write the appropriate instructions to create, if they do not exist, the following tables.

The first table is named "bikepoint" and has three attributes. The primary key is "ID", and it is an integer. The two attributes "Station" and "URL" are text strings with a maximum length of 50 and 100 respectively. The station attribute cannot be null.

The second table is called "manager" and has three attributes. The primary key is "code" while the foreign key is "codeBikepoint". Both are numbers. The name attribute is a string of up to 50 characters.

Relational schema

```
bikepoint(ID, stazione, url)
          stazione NOT NULL
```

```
manager(code, codeBikepoint, name)
        FK: codeBikepoint REFERENCES bikepoint.ID
```

2. Write a procedure `riempiDB(listBike)` which takes as input a list of three values and records them in the "bikepoint" table of the "DBbikesharing" database. It is assumed that the "listBike" list contains the values of ID, station and URL compatible with the domains of the attribute of the table. Manage any error in writing in the database by sending a warning message on the screen.

3. write the code of the "main" block in which the program

- calls the function as: `riempiDB([1, "Paravia", "http://www.tobike.it"])`
- asks the user to enter the three values needed to create a row of the manager table and write them into the table in the DB. Handle the error and warn the user in case of failure.

4. Write the code to print the result of the query to "show all the attributes of the bikepoint controlled by the manager with code 1"