

SIDBMS - 19 gennaio 2021  
CDLM Management e monitoraggio del turismo sostenibile

Prima parte: Teoria

Tempo: 30 minuti

Nome studente/ssa: \_\_\_\_\_ Matricola: \_\_\_\_\_

(IT) La prima parte della prova dura 30 minuti e vale 16 punti. Negli esercizi a risposta multipla, per ogni risposta corretta viene attribuito 1 punto e in caso di risposta errata verranno sottratti 0.5 punti. Non è consentito l'utilizzo del materiale didattico o di altro materiale.

(EN) The first part of the test lasts 30 minutes and is worth 16 points. In the multiple choice exercises, 1 point is awarded for each correct answer and 0.5 points will be subtracted in case of wrong answer. The use of teaching material or other material is not allowed.

Esercizio 1 (2 punti):

(IT) Data la seguente sequenza di istruzioni dire cosa stampa il programma e spiegare perchè.

(EN) Given the following sequence of instructions, write what the program print and explain why,

```
dato = [2+3,23]
copia = dato
copia[1] = 3
risultato = dato[0] > dato[1]
if risultato:
    print("Come volevasi dimostrare.")
else:
    print("Come?")
```

Esercizio 2 (3 punti)

| (IT) Considerando il tipo list e il tipo tuple in python:<br>(EN) Considering the list type and the tuple type in python. | VERO | FALSO |
|---|------|-------|
| Gli elementi della tupla possono essere aggiornati / The elements of the tuple can be updated                             |      |       |
| Gli elementi della lista possono essere aggiornati / The elements of the list can be updated                              |      |       |
| La lista può contenere una tupla ma non viceversa / The list can contain a tuple but not vice versa                       |      |       |

Esercizio 3 (3 punti):

(IT) Dato il seguente schema relazionale, scrivere le interrogazioni in SQL.

(EN) Given the following relational schema, write the queries in SQL.

```
Musician(IDT, Fullname, Nation)
Auditorium(IDD, Name, Address)
Performance(AMusician, AAuditorium, Date)
```

- a. Contare quanti turisti vengono dall'Italia / Count how many musicians come from Italy
- b. Elencare i nomi dei musicisti che si sono esibiti il "2018-10-03" / List the names of the musicians who performed on "2018-10-03"
- c. Contare quante performance si sono tenute per ciascun Auditorium / Count how many performances were held for each Auditorium

Esercizio 4 (2 punti)

(IT) Spiegare il concetto di lead nella Customer Relationship Management.

(EN) Explain the concept of lead In the Customer Relationship Management.

---

---

---

---

---

---

---

---

Esercizio 5 (3 punti).

| (IT) Un dato formato Json<br>(EN) A Json format   | VERO | FALSO |
|---|------|-------|
| Consente ad una persona di usare l'opera, di studiarla e di ricondividerlaPuò contenere un dato aperto / can contain an open data |      |       |
| deve essere generato soltanto tramite linguaggio Javascript / must be created only by using the javascript language               |      |       |
| Può essere deserializzato specificando l'intestazione / can be deserialized by specifying the header                              |      |       |

Esercizio 6 (3 punti)

(IT) Si vogliono gestire i dati di una catena di negozi di prodotti per vegetariani. Date il seguenti informazioni, progettare la base di dati con il modello E-R.

(EN) You want to manage the data of a chain of vegetarian products stores. Given the following information, design the database with the E-R model.

(IT) Ogni prodotto è definito dal nominativo, le calorie, la lista degli allergeni, può essere biologico oppure no, e può essere vegano oppure no. Un certo prodotto può essere disponibile in più negozi ad una certa quantità. Ciascun negozio ha un titolare, un numero di telefono ed un indirizzo. I negozi di grandi dimensioni sono specializzati e definiti “megastore”. Il megastore ha le caratteristiche del negozio normale e in aggiunta ha il reparto freschi, gestito da un certo responsabile di cui si vuole salvare il nominativo e che può gestire più reparti freschi. Il reparto freschi è inoltre caratterizzato dalla data di apertura e dalla categoria (“large” oppure “mini”).

(EN) Each product is defined by the name, the calories, the list of allergens, it can be organic or not, and it can be vegan or not. A certain product may be available in multiple stores at a certain quantity. Each shop has an owner, a telephone number and an address. Large stores are specialized and defined as “megastores”. The megastore has the characteristics of the normal shop and in addition it has the fresh department, managed by a certain manager whose name you want to save and who can manage several fresh departments. The fresh department is also characterized by the opening date and the category ("large" or "mini").

SIDBMS - 19 gennaio 2021  
CDLM Management e monitoraggio del turismo sostenibile

Seconda parte: Programmazione

Tempo: 1 ora e 30 minuti

Nome studente/ssa: \_\_\_\_\_ Matricola: \_\_\_\_\_

(IT) La seconda parte della prova dura 1 ora e 30 minuti e vale 16 punti. Eseguire gli esercizi su pycharm e copiare il risultato in questo form. Inviare le soluzioni dei tre esercizi all'indirizzo del docente entro il termine della prova. È consentito l'utilizzo del materiale didattico.

(EN) The second part of the test lasts 1 hour and 30 minutes and is worth 16 points. Perform the exercises on pycharm and copy the result into this form. Then send the solutions of the three exercises to the professor's address by the end of the test. The use of didactic material is allowed

**Esercizio 1 (4 punti)**

(IT) Riscrivere il codice utilizzando un ciclo for al posto del while e senza usare break pass e continue

(EN) Rewrite the code using a for loop in place of the while loop without using break pass and continue

```
v = [1, 2, 3, 4]
while True:
    if len (v) == 0:
        break
    v = v[:-1]
    if len (v) == 1:
        continue
    print (v)
```

## Esercizio 2 (7 punti)

(IT) Si vuole realizzare un programma che implementa un sondaggio per i visitatori di una località e salva i dati in formato Json. Creare nel proprio workspace il file sondaggio.json così composto:

(EN) You want to create a program that implements a survey for visitors to a location and saves the data in Json format. Create the survey.json file in your workspace as follows:

```
{
  [
    {
      "name": "Rossi Mario",
      "vote": 5,
      "text": "Very nice, but..."
    },
    {
      "name": "Bianchi Carlo",
      "vote": 4.5,
      "text": "All was perfect!"
    }
  ]
}
```

### 1. (1 punto)

(IT) Scrivere una funzione `startSurvey()` senza argomenti che chiede all'utente di inserire il proprio nome, un voto da 1 a 5 da attribuire alla visita, ed un testo libero. La funzione restituisce un dizionario con le chiavi "name", "vote", "text" contenenti i tre valori inseriti dall'utente.

(EN) Write the function `startSurvey()` without arguments that asks the user to enter their name, a vote from 1 to 5 to be attributed to the visit, and a free text. The function returns a dictionary with the keys "name", "vote", "text" containing the three values entered by the user.

### 2. (3 punti)

(IT) Scrivere una funzione:

`upgradeSurvey(answerDict, filename = "sondaggio.json")` che prende in ingresso un dizionario contenente le risposte della funzione `startSurvey` e il nome del file (usando `survey.json` come default). La funzione deserializza il dato Json presente nel file e aggiunge i dati contenuti nel dizionario passato come argomento, rispettando la struttura del file. La funzione quindi crea e salva il file Json aggiornato. La funzione restituisce una lista di due elementi contenente il numero di interviste presenti nel Json prima e dopo l'aggiunta. Ad esempio, aggiungendo un'intervista al file `survey.json` riportato sopra, la funzione deve restituire `[2, 3]`

(EN) Write the function:

`upgradeSurvey(answerDict, filename = "sondaggio.json")` that takes in a dictionary (containing the data returned by `startSurvey`) and the file name (using `sondaggio.json` as default). The function deserializes the Json data present in the file and adds the data contained in the dictionary passed as an argument, respecting the file structure. The function then creates and saves the updated Json file. The function returns a list of two elements containing the number of interviews present in the Json before and after the addition. For example, by adding an interview to the `sondaggio.json` file shown above, the function must return `[2, 3]`

3. (2 punti)

(IT) Scrivere una procedura:

`printAverage(filename="sondaggio.json")` che prende in ingresso il nome del file json da leggere (usando `survey.json` come default) e che stampa a video il valore medio di "vote".

(EN) Write a procedure:

`printAverage(filename = "sondaggio.json")` which takes in the name of the json file to be read (using `sondaggio.json` as default) and which prints the average value of "vote" on the screen.

4. (1 punto)

(IT) Scrivere un programma principale (con il costrutto `if __name__ == "__main__"`) che esegue in sequenza:

(EN) Write a main program (with the `if __name__ == "__main__"` construct) that executes in sequence:

```
printAvarage()  
upgradeSurvey(startSurvey())  
printAvarage()
```

### Esercizio 3 (5 punti)

(IT) Si vuole realizzare un sistema di gestione per il progetto Eco&Me. Il sistema prevede l'utilizzo del database mysql e del linguaggio python.

(EN) We want to create a management system for the Eco&Me project. The system requires the use of the mysql database and the python language.

#### 1. (2punti)

(IT) Scrivere le istruzioni per connettersi al DBMS mysql e, se non esiste, creare il database "ecoandme" ed usarlo. Scrivere le opportune istruzioni per creare, se non esistono, le tabelle descritte dal seguente schema relazionale.

(EN) Write the instructions to connect to the mysql DBMS and, if it does not exist, create the "ecoandme" database and use it. Write the appropriate instructions to create, if they do not exist, the tables described by the following relational schema.

```
guests(code, name, nationality)
  nationality DEFAULT "not given"

styles(id, name, description)
  name NOT NULL

tastes(aguest, astyle)
  FK: aguest REFERENCES guest.code
  FK: astyle REFERENCES styles.id
```

#### 2. (1 punto)

(IT) Scrivere il codice necessario per registrare dentro la tabella "styles" il contenuto della lista:

(EN) Write the code needed to record the contents of the following list in the table "styles" :

```
[123, "Hipster", "The 21st century subculture..."]
```

#### 3. (2 punti)

(IT) Scrivere la funzione `getStylesOf(code)` che prende in ingresso il "code" di un guest e restituisce la lista dei suoi stili salvati dentro "tastes".

(EN) Write the function `getStylesOf(code)` that takes a guest's "code" as input and returns the list of its styles stored in "tastes".