

ESERCIZIO BONUS 2 (+0.5) – Implementazione basi di dati con MongoDB

13/11/2018, Consegna: 21/11/2017, ore 23.55

Modalità di consegna → Attraverso la piattaforma di e-learning (<https://elearning-cds.unibo.it/course/view.php?id=29828>) di UNIBO. Passi da svolgere:

- 1) Collegarsi a: <https://elearning-cds.unibo.it/course/view.php?id=29828>, utilizzando le proprie credenziali istituzionali UNIBO (email/password) per l'accesso.
- 2) Scegliere il corso di Basi di dati della Laurea in Informatica per il Management, anno accademico 2018/2019 (codice: 70155).
- 3) Inserire la chiave di iscrizione: **bd2019**
- 4) Cliccare su Esercizio Bonus 2 → Consegna File, e procedere con l'upload del file contenente la soluzione dell'esercizio. E' possibile ripetere l'operazione di upload.
- 5) Quando si vuole consegnare l'elaborato DEFINITIVO, cliccare su "Consegna Compito". Da questo momento, nessun ulteriore upload è possibile. **Se non si clicca su "Consegna Compito", il docente NON vede l'elaborato.**
- 6) La sottomissione è possibile dal 13/11/2018 al 21/11/2018, ore 23.55.

COSA CONSEGNARE?

- Allegare un singolo file consegna2.js, contenente tutto il codice MongoDB/Javascript che implementa le specifiche dell'esercizio descritto a seguire.
-

PASSO PRELIMINARE (esercizio 0). In MongoDB, importare i documenti presenti nel file biblioteche.json (file presente sulla pagina Web del corso) all'interno del database culturaBO ed all'interno della collezione biblioteche. Il comando per importare i documenti è mongoimport, con questa sintassi:

```
mongoimport --db NOMEFILE --collection NOMECOLLECTION --file FILE_IMPORT
```

Ogni documento del file biblioteche.json dispone della seguente struttura:

```
{  name:      ****,  scheda_biblioteca:  ****,  descrizione:  ****,
  indirizzo:****,  telefono:  *****,  tipologia:  **** ,  quartiere:  ****,
  giornoChiusura: ****,  coordinates: **** }
```

(Alcuni campi potrebbero non essere presenti su tutti i documenti).

Il comando mongoimport deve essere eseguito da shell, e **NON deve figurare all'interno dello script consegna2.js.**

Implementare le seguenti funzionalità all'interno del file consegna2.js:

- 1) Contare il numero di documenti presenti nella collezione biblioteche. Stampare a video il seguente messaggio:
[PUNTO 1] Numero di biblioteche presenti: #NUM
dove #NUM è il risultato dell'operazione di conteggio.

- 2) Contare il numero di biblioteche presenti nel quartiere “Porto-Saragozza”:
[PUNTO 2] Numero di biblioteche : #NUM
dove #NUM è il risultato dell’operazione di conteggio.
- 3) Eliminare dalla collezione biblioteche tutti gli elementi con tipologia: “Biblioteca Universitaria”:
[PUNTO 3] Numero di biblioteche presenti dopo filtraggio: #NUM
dove #NUM è il numero di documenti presenti nella collezione biblioteche dopo aver effettuato l’operazione di rimozione.
- 4) Aggiornare la biblioteca di nome “Biblioteca R. Ruffilli”, aggiungendo le informazioni sui giorni di chiusura (Sabato e Domenica). Stampare a video il documento json modificato [usare la funzione printjson].
- 5) Le biblioteche possono disporre di una **bacheca** online. Sulla bacheca, le biblioteche possono inserire dei **POST relativi ad eventi mensili**. Ogni POST dispone di una data, un titolo, una descrizione. Ogni POST può essere commentato da utenti. I **COMMENTI** dispongono di testo e data. Gli **utenti** che scrivono commenti possono essere anonimi, oppure disporre di nome, cognome, età, mansione (i campi sono opzionali).
Modellare la bacheca nel database culturaBO, definendo le collezioni da creare. Popolare quindi la/e collezione/i con dati presenti nel file bacheca.txt (file presente sulla pagina Web del corso).
Usare **popolamento manuale** → sequenza di istruzioni insert.
[NOTA. Il numero di collezioni da creare è a scelta dello studente. In alternativa alla creazione di nuove collezioni, è possibile lavorare solo sulla collezione biblioteche aggiornandola opportunamente].
- 6) Scrivere la **query** che restituisce il numero di POST presenti nella bacheca della biblioteca “Biblioteca dell'Archiginnasio”, e relativi ad eventi di tipo: “Apertura straordinaria”, e stampa il seguente messaggio:

[PUNTO 6] Numero POST nella biblioteca dell’Archiginnasio: #NUM
- 7) Scrivere la **query** che restituisce il numero totale di COMMENTI utente presenti nella bacheca della biblioteca “Biblioteca dell'Archiginnasio”, e stampa il seguente messaggio:

[PUNTO 6] Numero COMMENTI nella biblioteca dell’Archiginnasio: #NUM

[NOTA. In JavaScript –come in Java- si può ottenere la dimensione di elementi in un array attraverso il campo length, ossia: nomeArray.length].

- 8) Scrivere la **query** che stampa a video nome, indirizzo e telefono (**solo questi campi**) delle biblioteche che hanno –nella loro bacheca- commenti scritti da utenti di età inferiore ai 30 anni o di professione “Studente”. Messaggio da stampare:

[PUNTO 7] Biblioteche recensite da studenti/giovani: #NOME -
#INDIRIZZO - #TELEFONO

- 9) Scrivere la **query** che restituisce la lista delle biblioteche (Nome/AnnoFondazione) la cui sede (campo **posizione**) è collocata nel raggio di 2 Km dal dipartimento di Informatica (latitudine: 44.497095, longitudine: 11.356001). Stampare a video nome ed indirizzo (**solo questi campi**) delle strutture selezionate, con il seguente messaggio:

[PUNTO 8] Biblioteche vicino al DISI: #NOME #INDIRIZZO

- 10) Scrivere la **query** che conta, per ogni quartiere, il numero di documenti presenti nella collezione biblioteche. Per ogni tipologia, stampare il messaggio:

[PUNTO 9] Quartiere: #NOME Numero Biblioteche presenti: #NUM

[VINCOLO: usare il comando di **aggregate** per eseguire la query.]

VINCOLI DI CONSEGNA

- L'implementazione della base di dati deve essere **completa** rispetto alle specifiche, e **funzionante**.
- **NON sono consentite consegne di gruppo. Consegne multiple** (ossia stesso codice da parte di più studenti) **NON sono valutate**.
- Inserire ove possibile dei **commenti** nel codice (in JavaScript i commenti si inseriscono –come in Java- tramite i caratteri speciali: //Commento su una riga o /* Commento su più righe */).
- **Seguire esattamente le specifiche del testo per quanto riguarda i nomi** del database, collezioni e campi etc.
- La consegna deve avvenire attraverso la piattaforma indicata nella prima pagina, ed entro la deadline stabilita. **Consegne via email NON saranno valutate**.
- L'assegnamento del **bonus** è una funzione booleana (assegnato/non assegnato).