

:mongoDB

א:

כתוב פונקציית הכנסה של המסמכים הבאים ל- collection.

```
id:0,Dep:Industrial engineering,age:50 Courses{math:95,database:7,algebra:14,}
id:1,Dep:CS,age:5 Courses{math:46,}
id:2,Dep:CS,age:29 Courses{math:91,database:21,algebra:60,}
id:3,Dep:Electrical Engineer,age:8 Courses{math:88,database:10,algebra:68,logic:33,}
id:4,Dep:Constructor,age:26 Courses{math:86,database:37,algebra:26,logic:95,history:32,}
id:5,Dep:Industrial engineering,age:10 Courses{math:87,database:11,}
id:6,Dep:Electrical Engineer,age:4 Courses{math:46,database:86,algebra:95,}
id:7,Dep:Industrial engineering,age:52 Courses{math:82,database:48,algebra:68,}
id:8,Dep:Constructor,age:53 Courses{math:23,database:47,algebra:93,logic:48,history:67,Chemistry:48,}
id:9,Dep:Industrial engineering,age:21 Courses{math:48,database:53,algebra:100,logic:22,}
id:10,Dep:Industrial engineering,age:39 Courses{math:96,database:93,algebra:62,}
id:11,Dep:Constructor,age:46 Courses{math:5,database:0,algebra:24,logic:63,}
id:12,Dep:CS,age:15 Courses{math:22,database:54,}
id:13,Dep:Constructor,age:13 Courses{math:82,database:67,}
id:14,Dep:Constructor,age:21 Courses{math:14,database:13,algebra:2,}
id:15,Dep:Electrical Engineer,age:35 Courses{math:66,database:41,algebra:64,logic:89,}
id:16,Dep:Electrical Engineer,age:25 Courses{math:18,database:77,algebra:44,logic:4,}
id:17,Dep:Electrical Engineer,age:38 Courses{math:67,database:26,algebra:86,logic:43,}
id:18,Dep:CS,age:49 Courses{math:53,database:48,}
id:19,Dep:Industrial engineering,age:59 Courses{math:4,database:76,algebra:0,}
```

```
var beginners =
[
  {
    "StudentId" : 1001,
    "StudentName" : "Steve",
    "age": 30
  },
  {
    "StudentId" : 1002,
    "StudentName" : "Negan",
    "age": 42
  },
  {
    "StudentId" : 3333,
    "StudentName" : "Rick",
    "age": 35
  },
],
db.students.insert(beginners);
```

דוגמא להכנסה מהירה:

ב:

כתבו פונקציה המחזירה ל Dep:CS, Electrical Engineer את הממוצע ציונים לכל מקצוע

דוגמא:

קלט:

id:16,Dep:Electrical Engineer,age:25 Courses{math:18,database:77,algebra:44,logic:4,}

id:17,Dep:Electrical Engineer,age:38 Courses{math:67,database:26,algebra:86,logic:43,}

id:18,Dep:CS,age:49 Courses{math:53,database:48,}

id:19,Dep:Industrial engineering,age:59 Courses{math:4,database:76,algebra:0,}

פלט:

Electrical Engineer{

Math:42.5

database :51.5

algebra:65

logic:23.5

}

CS:{

Math:53

database:48

}

Neo4j:

כתוב שאילתא שמוציאה את מספר האנשים שהם חברים של דני עד רמה 2 וגם ראו את כל הסרטים שהוא ראה או אהב.

elasticSearch:

עבור type של ספרים, בכל מסמך קיימים נתונים על הספר- שם, מחברים, ז'אנר הספר, הוצאה, שנת הוצאה ותקציר הספר-

1. כתוב פקודת הוספה של מסמך ל-index 'ספרים'
2. כתוב שאילתא שתחזיר את הספרים שהוצאו משנת 2000 והלאה ושז'אנר הספר הוא- 'מדע בדיוני' ושבתיקציר הספר יש את המילים – "Science Fiction" ו- "reality"

3. X-path:

נניח שקיים לנו קובץ XML המכיל ערים. כל עיר (city) מכילה attribute שם וקודקודים בנים המייצגים את כל מוסדות הלימוד שלה (institution) שבה. כל מוסד לימוד מכיל בתור attribute את מספר הלומדים במוסד(num).
כתוב שאילתא -path שתחזיר את שמות כל העיר שסך כל הלומדים במוסדות בעיר גדול ממליון

4. Stream4:

כתבו פונקציה אשר מקבלת מספר ומדפיסה את כל המספרים הראשוניים המחלקים אותו (בלי להשתמש בלולאות)

5. RDF:SPARQL

א. המר את הטבלה הבאה לטבלת RDF

ב. רשום שאילתא ב־SPARQL שמחזירה את שמות כל הנכדים של Dani

ID	name	age	Father_id
23	Dani	70	80
12	Michal	23	23
45	Yaron	49	67

6. TF-IDF:

נא דרגו את המשפטים הבאים – הראו או חישוב או מערכת שיקולים:

Q: Yael with Dani

A: Yael likes to go to the zoo with Yaron

B: Please go with my blue umbrella today, Thank you.

C: Yesterday Dani went to Yael.

D: Dani do you think to go with Michal?

E: I saw my neighbor, Dani, walk with his dog.