

[דף סיכום בבחינה](#)

מזהה בבחינה: 0212866586 מזהה סטודנט: 000000090648

מזהה קורס: 7217-180 שם קורס: DB Mongo

מספר שאלה	הערה	גיקוד מירבי	צ.ען
1.1		3.00	2.00
1.2		3.00	3.00
1.3		5.00	2.00
1.4		5.00	3.00
1.5		5.00	5.00
1.6		5.00	5.00
1.7		7.00	0.00
1.8		7.00	0.00
1.9		9.00	5.00
1.10		11.00	0.00
2.1		4.00	0.00
2.2		4.00	0.00
2.3		5.00	0.00
2.4		7.00	0.00
3	הגיקוד מאד מתחשב..	20.00	15.00

ציון בבחינה סופי : 40.00

הבחינה הבודקה בעמודים הבאים

**משרד העבודה
זרוע העבודה**
מה"ט המכוון הממשלתי להכשרה בטכנולוגיה ובמדע

הטכניון
מרכז טכנולוגי לישראל



בית הספר הארצי להנדסאים (ע"ר)

מחברת בחינה

הוראות לנבחן

(7)



ID 0212866586

Exam 000000090648



1. לפכי תחילת הבחינהanca קרא בעיון את ההוראות ומלא את הפרטים בכתב יד ברורה.
2. שם לב: מחברות הבחינה נסרקות למאגר נתונים. יש להקפיד: לא לкопל, לא לתרוש ולא לכתוב **בצבאים או בעפורים**.
3. כתוב את פרטייך האישיים על גבי מחברת הבחינה מיד עם תחילת הבחינה.
4. כתוב את תשובותיך בכתב יד ברור **משני צידי הדף**.
5. צין בראש כל עמודך (כולל הטוטה) במחברת זו. אל תכתוב דבר על דף אחר.
6. מתחקו אלכטוני ורשותם "מכוטל" על אותם חלקים שאינך רוצה שייבדקו.
7. אם מחברת זו לא תספיק לך, בקש מהמשגיח מחברת נוספת. כתוב על כל מחברת את פרטייך האישיים.
8. הוצאה מחברת בחינה (או חלק منهה) מוחוץ לחדר הבחינה הינה עבירה ממשמעית חמורה.
9. על הנבחן לצאתיה להוראות המשגיחים. כל פניה למשגיח תיעשה על ידי הרמת יד בלבד.
10. נבחן שיפרע למלהר הבחינה, יועד לדין ממשמעית של ביה"ס. ניתן להפסיק את בחינותו ולהתבעו בוועדת ממשמעת של ביה"ס.
11. נבחן לא יהיה רשאי לעזוב את חדר הבחינה, אלא על פי הנהלי בית הספר ובאישור המשגיח.
12. מותר לשחזר בשומר עוזר רק אם הדבר הותר במפורש בכתב גנוף שאلون הבחינה. הימצאות חומר אסור בידי הנבחן בשעת הבחינה על שולחנו, או בסטוף אליו, מהווה עבירה ממשמעת חמורה.

הערה: ההוראות מתיחסות לזכור ולנקפה אחד.

תאריך הבדיקה:

חתימת המרצה:

אנו מאשר כי קראתי את ההוראות האמורות לעיל:

חתימת הסטודנט



צין:

בהצלחה!

לתשומתך:

יש לכתוב את תשובותיך משני צידי הדף

אין לכתוב מעבר לקו האדום

יש לכתוב את הבחינה בעט (כחול/שחור) בלבד.

שם משפחה: *הירש* שם פרטי: *הירש*

מספר תעודה זהות: *1234567890* תאריך הבחינה: *20/07/2023*

מגמה: *1234567890* שם המרצה: *הירש* שם הקורס: *הנדסה אזרחית*

מועד: אמצע / א / ב / מיום סמסטר: חורף / אביב / קיץ



מחברת מס' *1234567890*



מתוך סה"כ **מחברות**

MongoDB

ההנחיות בשאלון זה מנוסחות בלשון זכר ומכוונות לנבחנות ולນבחנים כאחד.

בסיום המבחן חובה למסור למשגיחים את:

- א. מחברת הבחינה.
- ב. טופס הבחינה, עליו נכתב בכתב שם ות.ז. של הסטודנט.
- ג. שני דפי העזר שהביא הסטודנט למבחן (סעיף ג' בהמשך).

א. **משך המבחן:** שלוש שעות.

ב. **מבנה השאלון ופתח הערכה:** יש לענות על כל השאלות. הניקוד כתוב בגוף המבחן.

ג. **חומר עזר מותר בשימוש:**

שני דפי עזר, בגודל A4, שהינם **דפי עזר אישיים.**

חויבה לרשום את השם ות.ז. על הדפים האלו, ובסיום המבחן למסור אותם למשגיחים.

ד. **הוראות מיוחדות:**

- ד.1. כתוב יד שאיננו ברור - לא יבדק, או שיירד ניקוד משמעותי בגין כך!!
- ד.2. חל איסור מוחלט להוציא שאלון או מחברת בחינה מחדר הבחינה במהלך המבחן.
- ד.3. במידה ועל דפי הנוסחאות האישיים לא יופיעו פרטי הסטודנט - הדפים יוחרמו ולא יוכלו לשמש את הסטודנט במהלך המבחן.
- ד.4. יש לכתוב בעט בלבד.
- ד.5. אם נכתבה טיוטה במחברת הבחינה - יש למחוק אותה ע"י העברת קו כדי שלא תיבדק.
- ד.6. בשאלות בהן נדרש להסביר - חובה לנמק.
בשאלות שכאלו, תשובה לא מנומקת - לא תקבל ניקוד.
- ד.7. התחל לפטור כל שאלה בעמוד חדש במחברת הבחינה. כתוב בבירור את מספר השאלה.

בהצלחה!

לעוז
לעוז

MongoDB – מילויים

1. השאלה בחלק זה יבוססו על מסד נתונים במנג'ר המכיל מידע לגבי ספרים בספריה.
להלן דוגמא לאחד המסמכים שנמצא בטבלה שבמסד נתונים זה:

```
{  
    "_id": "978-3-16-148410-0",  
    "title": "The Great Story of all 48 Classes",  
    "description": "The auther presents readers with marvelous characters",  
    "cover_type": "Hardcover",  
    "author_name": "Lior B",  
    "book_dimensions": {  
        "width": 15,  
        "length": 10,  
        "height": 2.5  
    },  
    "best_seller": true,  
    "publication_year": 2024,  
    "publisher": "Ben Bassat Publishing Ltd",  
    "num_of_pages": 1759,  
    "language": "Hebrew",  
    "book_type": "Extreme Adventure",  
    "cost": 998.87,  
    "rating": 10  
}
```

כמובן, שלא כל המסמכים חייבים להכיל בתוכם את כל התוכנות שלעיל.

יש לקרוא היבט את כל הנדרש בכל שאלתה. כפי שהסביר בຄיתה, לאופן כתיבת השאלות יהיה חלק בניקוד שלהן, למשל:

- תחביר נכון. הקפדה על מספר סוגרים מדויק, כתיב מדויק בשמות פקודות, וכו'.
- כל "json internal" חייב להופיע בשורה נפרדת. למשל:

```
db.myCollection.aggregate(
```

```
[
```

```
    { $group: ... },  
    { $project: ... },  
    { $sort: ... },  
    { $match: ... }
```

```
]
```

- יש לדוק בשמות העמודות, לפי המוצג בדוגמה המסמך שבראש עמוד זה.
- כתוב ברור מאוד כפי שמצוין בפתח המבחן.

יש לכתוב את השאלות הנדרשות על **טבלה בשםם הפרי** המכילה מסמכים המתארים ספרים כמפורט לעיל:

א. (3 נקודות) כתוב שאלתה המציגת מידע על הספרים המוגדרים כ- **best_seller** **או** ככל שסוג הספר (**book_type**) הינו "History". המידע שנדרש להציג לכל אחד מהספרים האלו כולל **שם הספר, שם המחבר, תיאור הספר.**

ב. (3 נקודות) כתוב שאלתה המציגת את **כמויות** הספרים שפורסמו בשנת 2023 ואילך (**publication_year**), וגם סוג הכריכה שלהם (**"cover_type"** הינה "Paperback").
במילים אחרות, השאלת עונה על השאלה **"כמה ספרים כאלו ישנים.**
בסיוף זה עליך להשתמש בפקודת **find**.

ג. (5 נקודות) פטור את השאלתה הרצiosa בסעיף זה אמורה להציג את התשובה באופן הבא:
[{ 'Total number of books': 3 }]

כלומר יש להציג את התוצאה בשדה שמו "**Total number of books**", וرك השדה זהה אמרור להיות בפלט המתקיים מהשאלה, בדיק כמו שמצוג לעיל.

ד. (5 נקודות) לאחר מתן ציונים לדירוג הספרים (התוכנה rating), הוחלט לעדכן את מחירם של כל הספרים שערך הדירוג שלהם הינו הציון המקסימלי 10. לפי החלטה זו: מחיר הספר יעדכן להיות המקסימום שבין מחיר הספר הנוכחי לבין מחיר 50 ₪.
לדוגמה, אם מחירו של הספר לפני ניסיון העידכון הינו 49 ₪ אז המחיר יעדכן ל 50 ₪, ואילו אם מחירו היה 51 ₪ אז המחיר לא יעדכן ויישאר 51 ₪.

המשך בעמוד הבא

ה. (5 נקודות) מבין הספרים שפורסמו עד שנת 2021 (כולל), עבור הספר השלישי בעלותו (לא זה הכי יקר, אלא זה בעל המחיר השלישי בגודלו)- הציג רק את שמו של הספר והסוג שלו (book_type).

ניתן להניח שאין שני ספרים בעלי עלות זהה, וכי ישנו לפחות שלושה ספרים בספרייה.
יש לכתוב את השאלה הנדרשת בסעיף זה באמצעות שימוש בפקודה `find`.

לדוגמא, נניח שיש 4 ספרים שפורסמו עד שנת 2021:

ספר א' עולה 123 ₪,

ספר ב' עולה 98 ₪,

ספר ג' עולה 78 ₪,

ספר ד' עולה 101 ₪.

במצב זה הספר השני בגודלו מבחינת מספר העמודים הינו ספר ד' כי לאחר מינון לפי מספר העמודים מקבלים: 123, 98, 78 . יש להציג בק את שמו של הספר ואת הסוג שלו.

ו. (5 נקודות) פתרו את השאלה מהסעיף הקודם באמצעות שימוש בפקודה `aggregate`

ז. (7 נקודות) עבור כל הספרים שאינם כתובים בשפה האנגלית (כלומר לא כתובים באנגלית, אלא בשפה אחרת)- יש להזיל את מחירם ב 5 ₪ (מחירו הינו הערך שבשדי cost).

לדוגמא: נניח שקיים ספר בשפה האיטלקית שעלותו הינה 100.12 ₪, הפקודה הנדרשת בסעיף זה תתבקש לעדכן את עלותו ל- 95.12 ₪ ($100.12 - 5 = 95.12$). כך צריך לעדכן לכל אותם הספרים שלא כתובים בשפה האנגלית.

ח. (7 נקודות) בספריה שלו נכתבו ספרים בכל מיני שפות, ולכן ניתן לחשב את העלות הממוצעת של הספרים שנכתבו בכל שפה.
אם למשל נכתבו שני ספרים בשפה הולנדית, הראשון בעלות 100 ₪ והשני בעלות 200 ₪, הרי שהעלות הממוצעת של ספרים שנכתבו בשפה ההולנדית הינה 150 ₪.
כתב שאלתה שטרתה להציג את הערות הממוצעת של הספרים בכל שפה שבה יש ספרים בטבלה שלנו.
הפלט של השאלה צריך להציג את המידע באופן הבא (העריכים הם להמחשה):

[

{ 'Average cost': 534.43, 'Language Name': 'Hebrew' },

{ 'Average cost': 36.35727272727273, 'Language Name': 'English' }

]

מהדוגמה שלעיל ניתן להבין למשל כי הערות הממוצעת של ספרים בעברית- הינה 534.43 ₪.
כאמור, השאלה המבוקשת צריכה להציג את הערות הממוצעת של הספרים בכל אחת מהשפות.

ט. (9 נקודות) יש לכתוב שאלתה להציג השפה שבה נכתבוaci הרבה ספרים.
תשובות השאלתה צריכהות להציג את שם השפה זו ואת מספר הספרים שנכתבו בה.

. (11 נקודות)

נפח ספר מתקבל מתוצאת החישוב של המכפלת $width * length * height$.
כמוראה בהגדרת המסמך, מדוי הספר מופיעים בתוכנה: `book_dimensions`
כתוב שאלתה שמתירתה להציג את הנפח של הספר היקר ביותר שנכתב בכל אחת מהשפות.
הפלט הנדרש מהשאלתה צריך להיות ממוין בסדר יורד לפי אותו חישוב נפח, ולהיות מוצג במבנה
שכזה:

[

{ `_id`: 'Hebrew', 'Volume of the most expensive book': 375 },
{ `_id`: 'English', 'Volume of the most expensive book': 228.8 }

]

מהפלט של הדוגמה הנ"ל ניתן להבין למשל שהנפח של הספר היקר ביותר שנכתב בעברית הוא
.375

New York City

2. להלן חלק מסקריפט המגדיר מקומות ויחסים טיסות ביניהם:

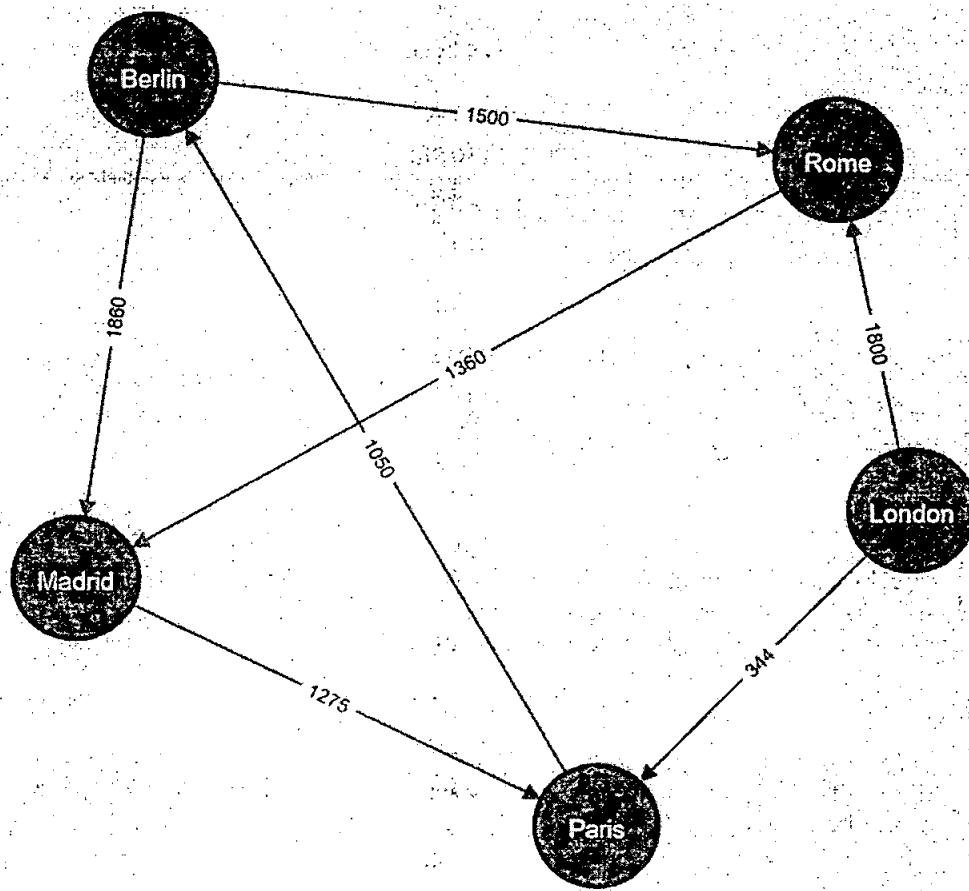
```
// Create city nodes
CREATE (london:City {name: 'London'})
CREATE (paris:City {name: 'Paris'})
CREATE (berlin:City {name: 'Berlin'})
CREATE (rome:City {name: 'Rome'})
CREATE (madrid:City {name: 'Madrid'})

// Create road relationships with distances and additional properties
CREATE (london)-[:Connected {distance: 344, cost:200, speedLimit: 65}]->(paris)
CREATE (paris)-[:Connected {distance: 1050, cost:210, speedLimit: 70}]->(berlin)
CREATE (berlin)-[:Connected {distance: 1500, cost:220, speedLimit: 130}]->(rome)
CREATE (rome)-[:Connected {distance: 1360, cost:230, speedLimit: 130}]->(madrid)
CREATE (madrid)-[:Connected {distance: 1275, cost:240, speedLimit: 60}]->(paris)
CREATE (london)-[:Connected {distance: 1800, cost:250, speedLimit: 130}]->(rome)
CREATE (berlin)-[:Connected {distance: 1860, cost:250, speedLimit: 130}]->(madrid)
```

הרצת הסקריפט מייצרת גרף המציג מידע על דרכים בין הערים הנתונות (מידע שאינו אמיתי).

יתכן בהחלה שיש יותר מסלול אחד בין שתי ערים.

- א. (4 נקודות) כתבו סקריפט להציג מספר הערים המופיעות במסלול הקצר ביותר (מבחינת מספר הקשנות) שבין הערים London ל- Madrid, ובנהנזה שהמסלול הזה מונה לכל יותר 10 קשנות. אדגיש: לא מדובר במסלול היכי קוצר מבינתי - distance שמדובר על רכיביו, אלא במסלול היכי קוצר מבוחנת מספר הקשנות שמרכיבותו אותן, והשאילתה נדרש להציג את מספר הערים במסלול.
- ב. (4 נקודות) כתבו סקריפט להציג סך הוצאות הכלולות של המסלול שהתקבל בסעיף ב' (כלומר סך כל הוצאות של כל קטעי המסלול הזה).
- רמז: שימו לב שעלי כל קשת מוגדרת התוכונה cost
- ג. (5 נקודות) כתבו סקריפט להציג שמות הערים שמהן ניתן להגיע בנסעה של שני שלבים אל העיר Madrid (העיר מדריד היא היעד).
- למשל, עבור המפה המוגדרת בעמוד הבא, ניתן לראות שהערים מהן ניתן להגיע למדריד בנסעה בת שני שלבים הם: פריז, לונדון וברלין.
- לא ניתן למשל להגיע למדריד בנסעה בת שני שלבים היוצאת מרומה.
- ד. (7 נקודות) הצג את שמות אותן הערים המקיימות את התנאים הבאים:
- יש ביניהם קשת ישירה לעיר מדריד.
 - המפרק על הקשת זהו קטן מ 2000
 - שמות הערים הללו מתחילה באות B



המשר בעמוד הבא

XML, DTD ו-XSD

3. נתן ה- DTD הבא:

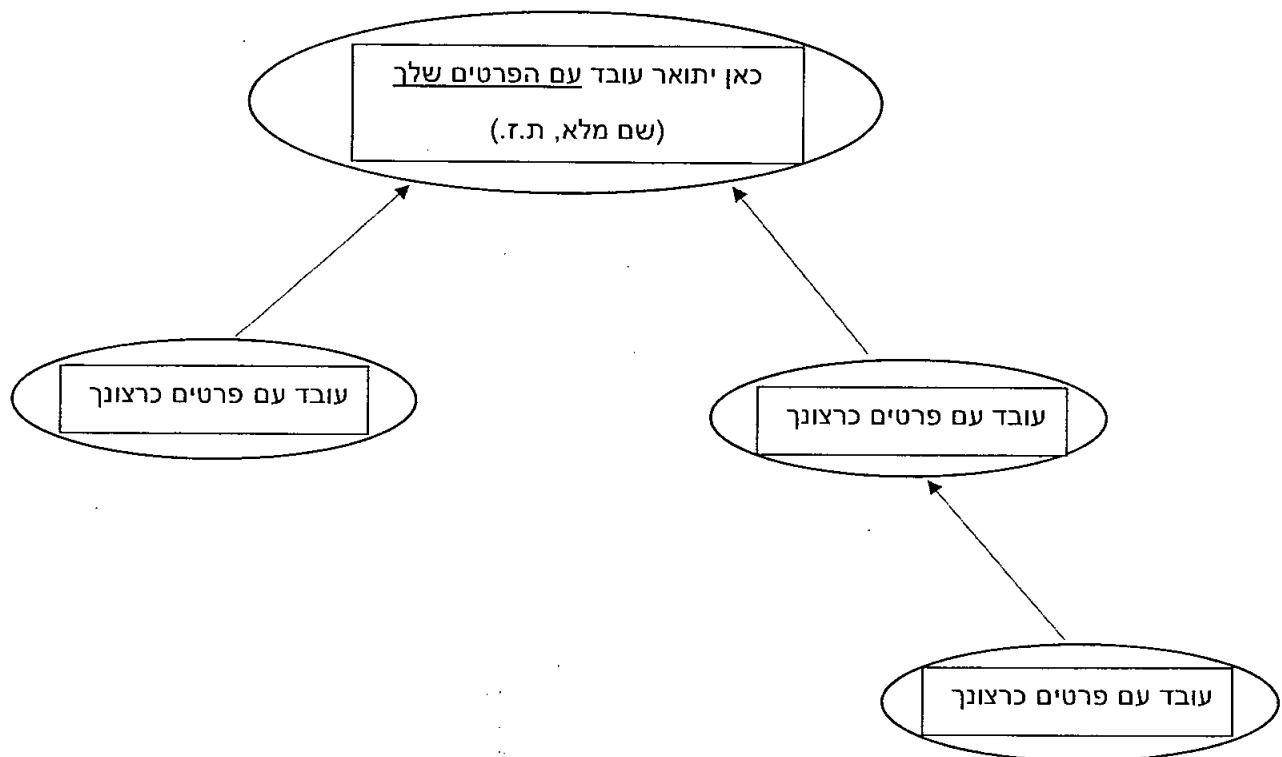
```
<!DOCTYPE Company [  
    <!ELEMENT Company (companyName,  
                      managerName,  
                      employees  
                    )>  
  
    <!ELEMENT companyName (#PCDATA)>  
    <!ELEMENT managerName (#PCDATA)>  
    <!ELEMENT employees (employee+)>  
  
    <!ELEMENT employee (empName,salary, empAddress+)>  
    <!ELEMENT empName (last, middle?, first)>  
    <!ELEMENT first (#PCDATA)>  
    <!ELEMENT middle (#PCDATA)>  
    <!ELEMENT last (#PCDATA)>  
    <!ELEMENT salary (amount, currency)>  
    <!ELEMENT amount (#PCDATA)>  
    <!ELEMENT currency (#PCDATA)>  
    <!ELEMENT empAddress (#PCDATA | city | street | houseNum)*>  
    <!ELEMENT city (#PCDATA)>  
    <!ELEMENT street (#PCDATA)>  
    <!ELEMENT houseNum (#PCDATA)>  
  
    <!ATTLIST Company companyId ID #REQUIRED>  
    <!ATTLIST employee empId ID #REQUIRED>  
    <!ATTLIST employee managerId IDREF #IMPLIED>  
    <!ATTLIST employee type (FullTime | Student ) #REQUIRED>  
    <!ATTLIST employee managerOf IDREFS #IMPLIED>  
]>
```

הסביר לגבי התקונה managerOf שבאלמנט employee : בתוכנה זו יש את ה- ID של המנהל של אותו employee.

הסביר לגבי התקונה managerOf שבאלמנט employee : בתוכנה זו יש את כל ה- ID של כל עובדים שכפויים לאותו employee.

המשך בעמוד הבא

(20 נקודות) בהתאם ל- DTD שלעיל. כתבו XML המכיל 4 עובדים במבנה הארגוני הבא, כך שהוא יקיים את הדרישות המינוחדות בהמשך עמוד זה:



דרישות מינוחדות בכתיבה ה- XML :

- אתה מוגדר כמנהל של הארגון ששורטט לעיל, וכן עלייך להכנס במקומות המתאימים את ת.ז. שלך ואות שמו המלא.
- הקפד שניים מתוך ארבעת העובדים בשרטוט הנ"ל - יוגדרו במשרת סטודנט, והשניים האחרים יוגדרו כעובדים במשרה מלאה (FullTime).
- הקפד להגדיר כל כתובת (empAddress) במבנה אחר, לפי המותר ב- DTD.

בהצלחה,

ליאור.


```
<day>13</day>

<month>July</month>

<year>2024</year>

</date>

<items>

<item serialNum="5004" discountedItems="5006">

    <itemDesc>Laptop</itemDesc>

    <itemPrice>1200</itemPrice>

    <quantity>1</quantity>

    </item>

    <item serialNum="5005">

        <itemDesc>Mouse</itemDesc>

        <itemPrice>20</itemPrice>

        <quantity>2</quantity>

        </item>

    </items>

</order>

</orders>
```

ונמצאים בשימוש נוכן, לפי ה-DDT. לדוגמה, האלמנט middleName נדרש להיות תקין אם הוא מופיע, גם אם הוא אופציוני.

מבחן סדרה של פעולות:

אשונה אם מבקשים רק אוכלוסייה מסוימת.

בנייה נבוקש לקבל רק שמות מסוימים

לישת ובקש למן לפי גיל בסדר עידן.

כיעית מילג על המבוגר ביותר (לפי דרישת המרגיל).

לראות שהובילו את אותו התוצאות כמו בהרואה של ה-phi.

```
Classmate> devkit:64\src>gdb -q
...
...
[1] $watch(1) = {0}
[2] $watch(2) = {0}
[3] $watch(3) = {0}
[4] $watch(4) = {0}
[5] $watch(5) = {0}
[6] $watch(6) = {0}
[7] $watch(7) = {0}
[8] $watch(8) = {0}
[9] $watch(9) = {0}
[10] $watch(10) = {0}
[11] $watch(11) = {0}
[12] $watch(12) = {0}
[13] $watch(13) = {0}
[14] $watch(14) = {0}
[15] $watch(15) = {0}
[16] $watch(16) = {0}
[17] $watch(17) = {0}
[18] $watch(18) = {0}
[19] $watch(19) = {0}
[20] $watch(20) = {0}
[21] $watch(21) = {0}
[22] $watch(22) = {0}
[23] $watch(23) = {0}
[24] $watch(24) = {0}
[25] $watch(25) = {0}
[26] $watch(26) = {0}
[27] $watch(27) = {0}
[28] $watch(28) = {0}
[29] $watch(29) = {0}
[30] $watch(30) = {0}
[31] $watch(31) = {0}
[32] $watch(32) = {0}
[33] $watch(33) = {0}
[34] $watch(34) = {0}
[35] $watch(35) = {0}
[36] $watch(36) = {0}
[37] $watch(37) = {0}
[38] $watch(38) = {0}
[39] $watch(39) = {0}
[40] $watch(40) = {0}
[41] $watch(41) = {0}
[42] $watch(42) = {0}
[43] $watch(43) = {0}
[44] $watch(44) = {0}
[45] $watch(45) = {0}
[46] $watch(46) = {0}
[47] $watch(47) = {0}
[48] $watch(48) = {0}
[49] $watch(49) = {0}
[50] $watch(50) = {0}
[51] $watch(51) = {0}
[52] $watch(52) = {0}
[53] $watch(53) = {0}
[54] $watch(54) = {0}
[55] $watch(55) = {0}
[56] $watch(56) = {0}
[57] $watch(57) = {0}
[58] $watch(58) = {0}
[59] $watch(59) = {0}
[60] $watch(60) = {0}
[61] $watch(61) = {0}
[62] $watch(62) = {0}
[63] $watch(63) = {0}
[64] $watch(64) = {0}
[65] $watch(65) = {0}
[66] $watch(66) = {0}
[67] $watch(67) = {0}
[68] $watch(68) = {0}
[69] $watch(69) = {0}
[70] $watch(70) = {0}
[71] $watch(71) = {0}
[72] $watch(72) = {0}
[73] $watch(73) = {0}
[74] $watch(74) = {0}
[75] $watch(75) = {0}
[76] $watch(76) = {0}
[77] $watch(77) = {0}
[78] $watch(78) = {0}
[79] $watch(79) = {0}
[80] $watch(80) = {0}
[81] $watch(81) = {0}
[82] $watch(82) = {0}
[83] $watch(83) = {0}
[84] $watch(84) = {0}
[85] $watch(85) = {0}
[86] $watch(86) = {0}
[87] $watch(87) = {0}
[88] $watch(88) = {0}
[89] $watch(89) = {0}
[90] $watch(90) = {0}
[91] $watch(91) = {0}
[92] $watch(92) = {0}
[93] $watch(93) = {0}
[94] $watch(94) = {0}
[95] $watch(95) = {0}
[96] $watch(96) = {0}
[97] $watch(97) = {0}
[98] $watch(98) = {0}
[99] $watch(99) = {0}
[100] $watch(100) = {0}
```

Note that in the above example:

(*) The empty match criteria indicates that we are observing all the documents.

(* We are grouping according to the value of the semester number. Mean grouping is based on the SemesterNum column.

(*) We calculate the sum of ages of all the students in each group.

(*) The calculation result is presented under the column name "total" (in coming example we will see how to change that name).

1. היעדר שדה **middleName** (שם אמצעי): לפי ה-DTD, צריך להיות אלמנט **middleName** בתוך **.custName**.

2. בקובץ ה-XML השדה זהה קיימן אך ריק, וזה עשוי להיחשב כשגיאה אם שם אמצעי נדרש או שהשדה לא אמרו להיות ריק.

3. מבנה כתובת לא עקבי: בקובץ ה-XML נעשה שימוש באלמנט **houseNum** בתוך **address**, שמצוין גם ב-DTD. יש לוודא שהשימוש באלמנט זהה ובמבנה הכלול של הכתובת הוא עקבי ומתחאים למפרט ב-DTD.

4. ערך לא תקין עבור אמצעי התשלום: לפי ה-DTD, הערך החוקיים לattributο paymentMethod הם "Cash", "Points" או "CreditCard". בקובץ ה-XML מופיע הערך "MyPoints", שאינו תואם את הערך שהוגדרו ב-DTD.

5. שימוש באטሪビוט **discountedItems**: לפי ה-DTD, האטሪビוט **discountedItems** נמצא אלמנט **item** והוא לא חובה (#IMPLIED). יש לוודא שהוא נמצא בשימוש נכון ואם נדרש, יש להסתור.

אלמנטים ואטሪビוטים אופציונליים לא נכוןים/שגוים: יש לוודא שכל האלמנטים והאטሪיבוטים האופציונליים

CREATE (chicago)-[:FLIGHT {distance: 802, duration: 130}, pilot: Linda Clark]->(dallas) CREATE (cleveland)-[:FLIGHT {distance: 462, duration: 95}, pilot: Chris Johnson]->(nyc) CREATE (nyc)-[:FLIGHT {distance: 1389, duration: 210, pilot: David Kim}]->(chicago) CREATE (la)-[:FLIGHT {distance: 1745, duration: 270, pilot: Chris Johnson}]->(chicago) MATCH (c1:City)-[:FLIGHT]->(f:FLIGHT)->(c2:City) RETURN COUNT(f) AS chrisJohnson_flights MATCH (c1:City)-[:FLIGHT {pilot: Chris Johnson}]->(c2:City) Task 5: Find the Total Number of Flights Piloted by "Chris Johnson" LIMIT 1
ORDER BY f.distance ASC RETURN f.distance AS distance >(dallas:City)-[:FLIGHT]->(c1:City) MATCH (la:City {name: 'Los Angeles'})-[:FLIGHT]->(c1:City) Task 4: Find the Shortest Flight from Los Angeles to Dallas RETURNS COUNT(f) AS total_flight_routes MATCH (c1:City)-[:FLIGHT]->(c2:City) Task 3: Return the Total Number of Flight Routes LIMIT 1
ORDER BY f.distance DESC RETURN c1.name AS from, c2.name AS to, f.distance MATCH (c1:City)-[:FLIGHT]->(c2:City) Task 2: Find the Shortest Flight and Return its Distance LIMIT 2
WHERE f.duration RETURN DISTINCT f.pilot Task A: Create an XML Document Based on the P The XML document should adhere to the DTD it should contain 5 elements (or "fields") that match the provided structure. Example XML Document: Based on the DTD, a ve cus <order transactionId= paymethod="MyP <customer customerId= <cus </cus <firstN <middleName> <lastName>Doe</lastN <city>Los Angeles</ci <street>123 Main Street</stre <houseNum>456</hou </cu;

```

$skip   •
db.orders.aggregate([
  { $sort: { date: -1 } },
  { $skip: 10 }
]);
3. Indexing
createIndex(keys, options) •
db.users.createIndex({ name: 1 });
dropIndex(indexName) •
db.users.dropIndex("name_1");
listIndexes() •
db.users.listIndexes();
dropIndexes() •
db.users.dropIndexes();
6. Data Validation
createCollection(name, options) •
db.createCollection("users", {
  validator: {
    $jsonSchema: {
      bsonType: "object",
      required: ["name", "email"],
      properties: {
        name: {
          bsonType: "string",
          description: "must be a string and is required"
        },
        email: {
          bsonType: "string",
          pattern: "^.+@.+\\..+$",
          description: "must be a string and match the email format"
        }
      }
    }
});

```

7. Utility Methods

```

renameCollection(newName, options)
db.users.renameCollection("customers");

```

```

drop()
db.users
stats()
db.users
.db
.find( {"cover_type":{$exists:true} },{book_name:1, first_name:1, num_of_pages:1} )
.updateMany( { $and: [ { "Language": "English" }, { "rating": {$lte: 4.3} } ] } , {$set: {"cover_type": "Fiction", "is_popular": false} })
.aggregate( [ { $match: { "publication_year": {$gte: 2010, $lte: 2018} } } ], { $group: { "number of books": {$sum: 1} } } , { $project: { _id: 0, "number of books": 1 } } )
.find( { $and:[ {"publication_year":{$gte:2010}}, {"publication_year":{$lte:2018}} ] } ) , { $group: { "number of books": {$sum: 1} } } , { $project: { _id: 0, "number of books": 1 } } )

```

Neo4j Script to Create Nodes and Relationships

Create City Nodes:

```
CREATE (nyc:City {name: 'New York'})
```

```
CREATE (la:City {name: 'Los Angeles'})
```

```
CREATE (chicago:City {name: 'Chicago'})
```

```
CREATE (dallas:City {name: 'Dallas'})
```

```
CREATE (cleveland:City {name: 'Cleveland'})
```

Create Flight Relationships with Distances and Additional Properties:

```
CREATE (nyc)-[:FLIGHT {distance: 2451, duration: 30 min}]-> (chris:Person {name: 'Chris Johnson'})
```

```
CREATE (nyc)-[:FLIGHT {distance: 790, duration: 12 min}]-> (jane:Person {name: 'Jane Smith'})
```

```
CREATE (la)-[:FLIGHT {distance: 1235, duration: 25 min}]-> (nancy:Person {name: 'Nancy Wilson'})
```

1. CRUD Operations

Braa

db.BraaCollection.find({\$or:[{
 "best_seller":{\$exists:true},
 "book_type":
 {"\$in":["History"]},
 "video.title":{\$exists:true},
 "description":
 {"\$exists:true},
 "author.name":{\$exists:true}}]}))

(1.1)

db.BraaCollection.find({\$and:[{
 "publication.year":{\$gt:2023},
 "cover_type":
 {"\$in":["Paperback"]}}]}))

"publicaton.year":{\$gt:2023},
"cover_type":
 {"\$in":["Paperback"]}}]}))

.count()

db.BraaCollection.aggregate([
 {"\$match":{
 "\$and":{
 {"publication.year":
 {\$gt:2023}},
 {"cover_type":
 {"\$in":["Paperback"]}}}}},
 {"\$group":{
 "_id":
 {"\$concat":
 ["Total",
 "Number of books"]},
 "sum":{\$sum:1}}},
 {"\$project":
 {"id":0,"Total Number of Books":1}}])

היה
צדי
להנדי^ר
פ^ר
הנדי^ר
הנדי^ר
הנדי^ר
הנדי^ר
הנדי^ר

db.BraaCollection.aggregate([
 {"\$match":{
 "\$and":{
 {"rating":10},
 {"cost":{\$gt:100}},
 {"cost":{\$lt:50}}}}},
 {"\$group":{
 "_id":
 {"\$concat":
 ["Max",
 "Cost"]},
 "sum":{\$sum:1}}},
 {"\$project":
 {"id":0,"Max Cost":1}}])

היה נדרש להשתמש באופרטור MAX



5

(1.5)

-7

db.BraaCollection.find({\$or: [{\$and: [{"cost": {"\$gt": 0}}, {"year": {"\$gt": 1980}}]}], {cost: 1, id: 1, book_type: 1}).sort({cost: 1}).skip(5).limit(3)

צורך "לכבות" את pi
(לא הורדתי ניקוד)

db.BraaCollection.aggregate([{\$match: {"publication_year": {"\$gt": 1980}}}, {"\$sort": {"cost": 1}}, {"\$skip": 5}, {"\$limit": 3}])

צורך "לכבות" את pi
(לא הורדתי ניקוד)

5
(1.6)

-5E

0
(1.8)

0
(1.7)

db.BraaCollection.aggregate([
{\$group: {
_id: "\$language",
\$sum: 1
}}
{\$sort: {_id: -1}},
{\$limit: 13},
])

נדרש היה גם להוסיף project

5
(19)

תיקוד התרגילים זהה זהנו 7 נקודות מתוך 9
הורזתי 2 נקודות נוספת עד כיו שברוב הטעופים שלמעלה לא הקפדה על מה שם כתוב
בהתראות המבחן גם לדיבורתי על זה רצבות בכתיבת: לא "לשבר שורות" באמצעות "internal json"
לכן בסכ"ה ניקוד הסעיף הזה יהיה 5 נקודות.

0
(1.10)

להלן ריבס בפתרון זהה הינט בלתי קרייאם. לפחות אפשר לכתוב מרווח יותר. יש לנושא

<Company>

התיחסות בהוצאות המבחן אף דיברנו עלן וובות בכיתה

<~~Company~~ Name>

<ManagerName>

<Employees>

managers

<Employee> Name = Brack Higa

salary = 1000 empAddress = > first

empName = Brack Higa = "Brack"

first = "Bracey">

<first> Brack / first >

<last> Higa / last >

<Employee>

<salary> salary = "1000" </salary>

<currency> currency = "USD" </currency>

</salary>

<empAddress>

<city> K City >

<street> 14 Street >

<houseNum> 14 / houseNums

<shopAddress>

<Company> contactID = "111" >

contactID = 111 = "919365567" >

<ManagerID> ID = "919365567" >

~~Employee FullTime~~ >

אלו לא אלמנטים אלא תכונות.

~~ManagerOf~~ "3"

בנוסף יש טעויות תחביר

~~Employee~~ >

~~Employee~~ >

~~Employee~~ >

~~First~~ > Those / First >

~~Last~~ - Hoyz / Last > ~~Employee~~ Names >

~~Salary~~ >

~~Amount~~ > 1000 ~~Amounts~~ >

~~Currency~~ > 500 ~~Currencies~~ >

~~Salary~~ >

~~EmployeeAddress~~ >

~~City~~ k ~~City~~ >

~~Street~~ > 3 ~~Streets~~ >

~~HouseNumber~~ > ~~HouseNames~~ >

~~EmployeeAddress~~ >

~~CompanyID~~ ID = "5" >

אלו לא אלמנטים

~~EmployeeID~~ ID = "1" >

לא תכונות.

~~EmployeeAddress~~ > ManagerID =
10960565

~~EmployeeType~~ Type = "FullTime" >

~~Employed~~ ✓

EMPRESA

~~(First) \Rightarrow (LHS) \neq RHF~~

Just > Book first > Group Name

Oppression

~~Session 1002/1 month~~

~~SECRETARY OF DEFENSE~~

(Sd) (Avy)

~~REMOVED ADDRESS
CCITY~~

~~ESTABLISHED FEB 12~~

change. New > 82 / house & 100%

(~~Temp Address~~)

COMPAGNA COMPAGNIA D'IS

Employed people = "3" >

קשה לקרו

~~Employment record 1993 baby~~

Complex types = "Structs"

~~Playground~~

או הגדה בכתיבת תכונות לעמודות אלמנטים.

~~employees~~

~~U.S. Company~~

סימנתי רה חלק מהמחלומות הנ"ל.

לא כל התוכנות מומומשות בהצלחה בקood שכתבת.

יש אידוק במימוש של תוכנות לעומת אלמנטים.

לא מומושה בהצלחה ההייכקה של העובדים כפי שנדרשה בשאלת, ויש טעויות
במימוש
.managerOf

לגביו הכתב-התיחסתי למעלה, היה ניתן לפחות לדעوه מעט בין השורות.

הניקוד מאוד מתחשב..

15

(3)

0

(2.1)

0

(2.2)

0

(2.3)

0

(2.4)

