

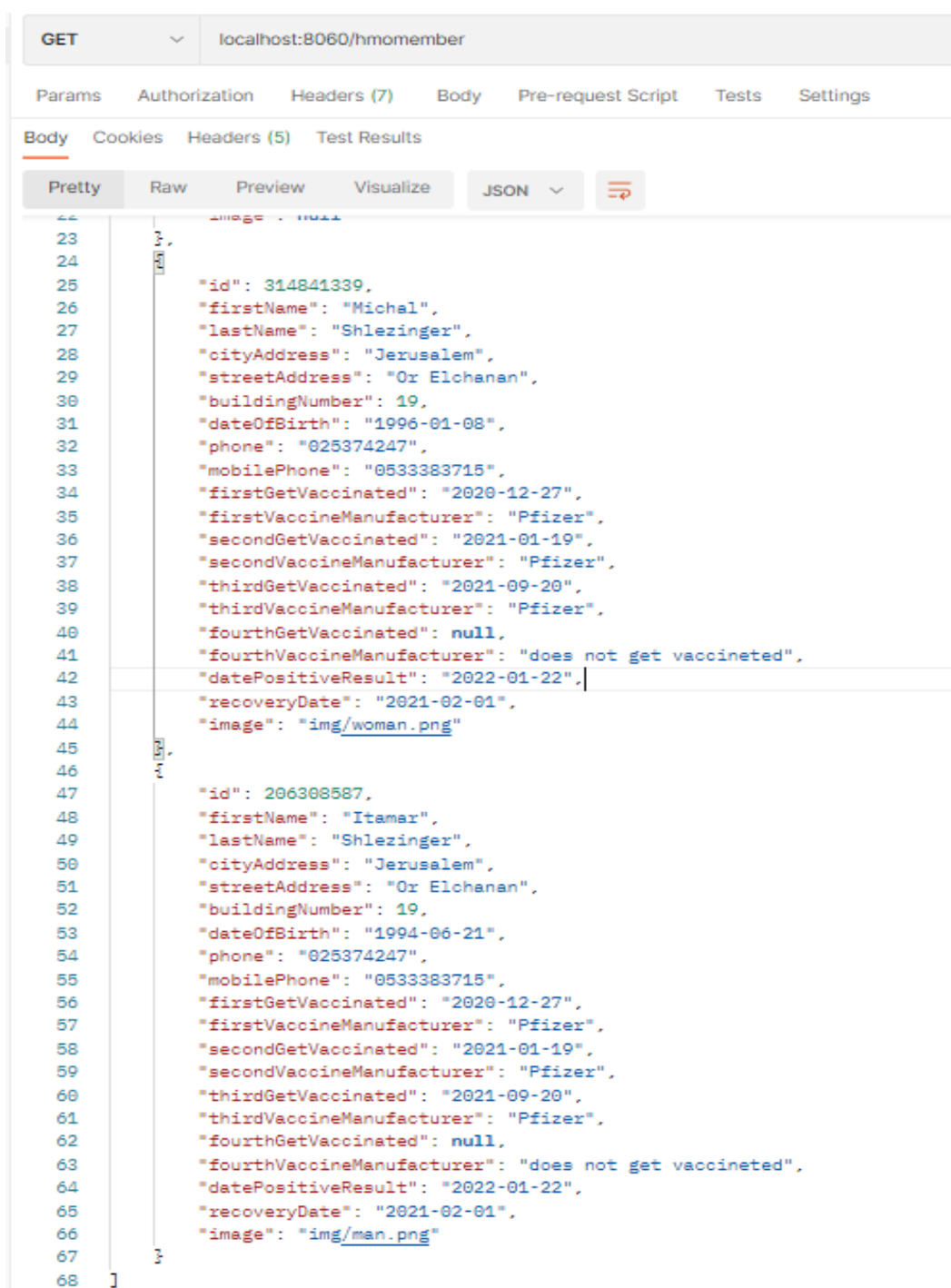
תרגיל בית הדסים

מיכל שלזינגר 314841339

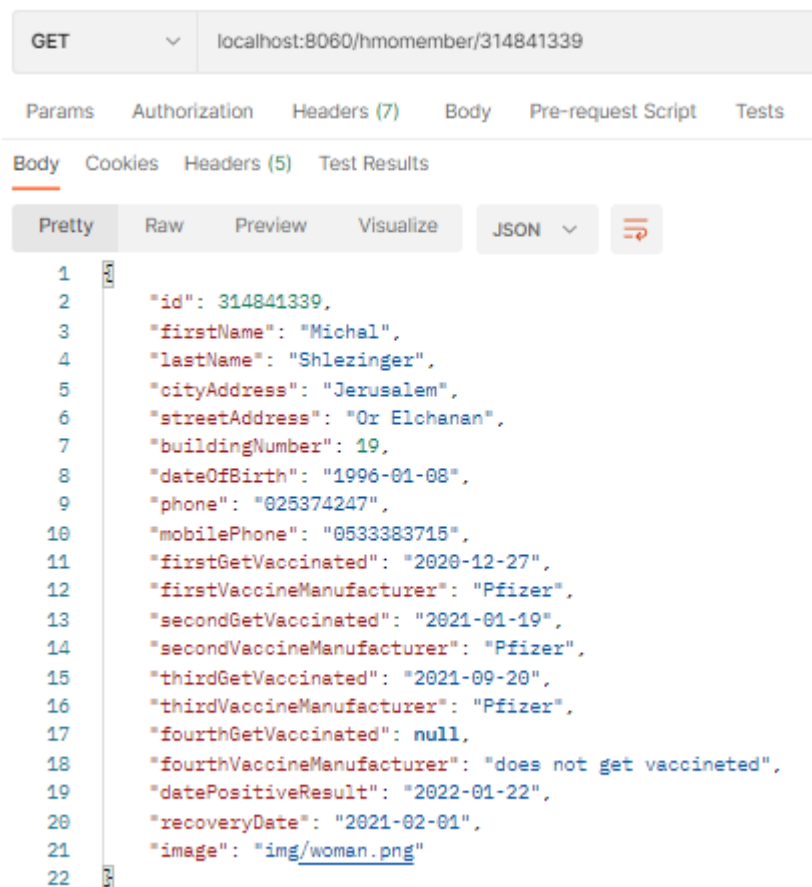
בפרויקט זה מימשתי צד השרת הכולל שליפת נתונים והכנסה רשומות למאגר מידע של חברי קופת חולים.

לצורך תצוגת הנתונים השתמשתי באפליקציית postman - היא אפליקציה שמאפשרת לנו לעשות בדיקות Api

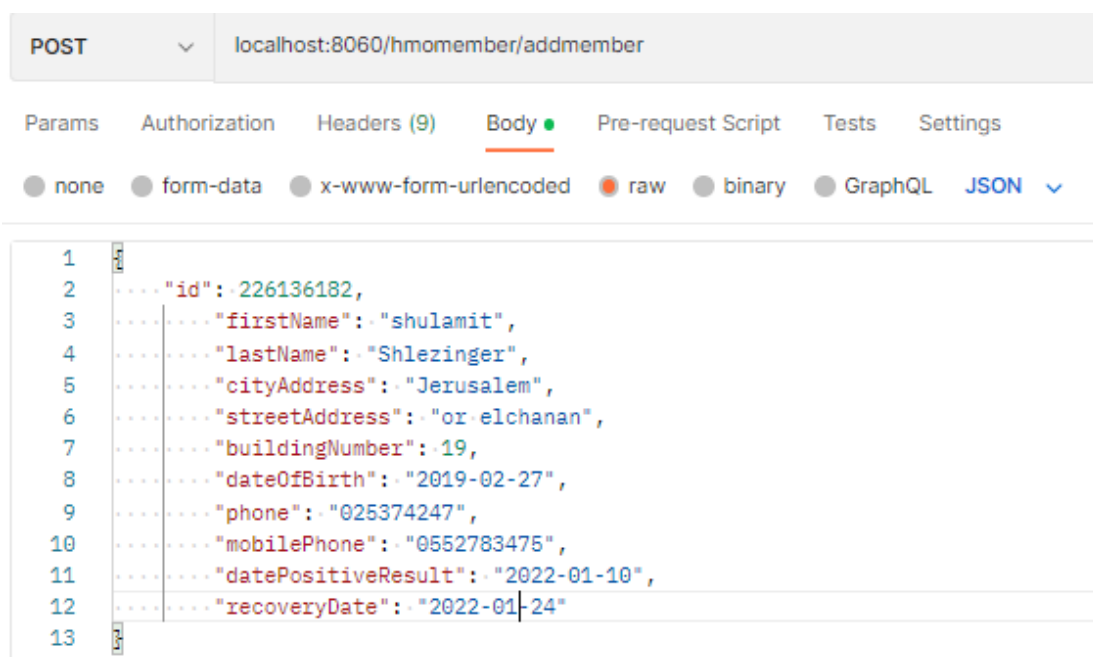
במסך הבא המערכת מציגה את כל החברים בקופת חולים -



כדי להציג פרטים של חבר מסוים ננתב לתעודת זהות שלו
-localhost:8060/hmomember/{{id}}



ע"מ להוסיף חבר חדש לקופת חולים ננתב ל
-localhost:8060/hmomember/addmember
ששם יש את כל השדות
הנדרשים למילוי, לא את כולם חייבים למלאות....



דבר ראשון נוודא שהתעודת זהות לא קיימת ואם קיימת כבר תוצג שגיאה-

```

1  {
2    "timestamp": "2023-05-11T11:47:38.120+00:00",
3    "status": 500,
4    "error": "Internal Server Error",
5    "message": "HMO Member with id exists",
6    "path": "/hmomember/addmember"
7  }

```

פרטים שלא מולאו - כמו אחד או יותר מהחיסונים יוכנס למישתנה של הסוג
יצרן של החיסון "שלא התקבל חיסון" -
דוגמא איך נראה פרטי חבר חדש בטבלה -

```

GET localhost:8060/hmomember

Params Authorization Headers (7) Body Pre-request Script Tests Settings
Body Cookies Headers (5) Test Results

1  [
2    {
3      "id": 226136182,
4      "firstName": "shulamit",
5      "lastName": "Shlezinger",
6      "cityAddress": "Jerusalem",
7      "streetAddress": "or elchanan",
8      "buildingNumber": 19,
9      "dateOfBirth": "2019-02-27",
10     "phone": "025374247",
11     "mobilePhone": "0552783475",
12     "firstGetVaccinated": null,
13     "firstVaccineManufacturer": "does not get vaccinated",
14     "secondGetVaccinated": null,
15     "secondVaccineManufacturer": "does not get vaccinated",
16     "thirdGetVaccinated": null,
17     "thirdVaccineManufacturer": "does not get vaccinated",
18     "fourthGetVaccinated": null,
19     "fourthVaccineManufacturer": "does not get vaccinated",
20     "datePositiveResult": "2022-01-10",
21     "recoveryDate": "2022-01-24",
22     "image": null
23   },
24   {
25     "id": 314841339,
26     "firstName": "Michal",
27     "lastName": "Shlezinger",
28     "cityAddress": "Jerusalem",
29     "streetAddress": "Or Elchanan",

```

אם נכניס תאריך החלמה לפני תאריך חשיפה נקבל שגיאה -

```

1  {
2    "id": 123456789,
3    "firstName": "shulamit",
4    "lastName": "Shlezinger",
5    "cityAddress": "Jerusalem",
6    "streetAddress": "or elchanan",
7    "buildingNumber": 19,
8    "dateOfBirth": "2019-02-27",
9    "phone": "025374247",
10   "mobilePhone": "0552783475",
11   "datePositiveResult": "2022-01-10",
12   "recoveryDate": "2021-01-24"
13 }
14

```

body Cookies Headers (4) Test Results

Pretty Raw Preview Visualize JSON ↕

```

1  {
2    "timestamp": "2023-05-11T11:33:43.875+00:00",
3    "status": 500,
4    "error": "Internal Server Error",
5    "message": "The date of the recovery is incorrect because is before the date of positive result",
6    "path": "/hmomember/addmember"
7  }

```

בדיקה שהשם פרטי, שם משפחה, עיר ורחוב שמתקבלים כל התווים הם אותיות ואם לא יוצג שגיאה לדוגמא -

Body Cookies Headers (4) Test Results

Pretty Raw Preview Visualize JSON ↕

```

1  {
2    "timestamp": "2023-05-11T11:42:24.761+00:00",
3    "status": 500,
4    "error": "Internal Server Error",
5    "message": "The first name and the last name must to be all letters",
6    "path": "/hmomember/addmember"
7  }

```

בדיקה שהטלפונים אכן מכילים רק 10 ספרות וכולם ספרות ובמקרה שלא תוצג שגיאה -

Pretty Raw Preview Visualize JSON ↕

```

1  {
2    "timestamp": "2023-05-11T11:51:12.531+00:00",
3    "status": 500,
4    "error": "Internal Server Error",
5    "message": "The mobile number is invalid",
6    "path": "/hmomember/addmember"
7  }

```

דוגמא בסיסית לאיך יראה דף הוספה בצד לקוח (בלי חיבור לצד שרת..)

First Name:

Last Name:

ID:

City Address:

Street Address:

Building Number:

Date Of Birthday:

Phone:

Mobile Phone:

Image לא נבחר קובץ

מידע לגבי התקנות:

מאגר המידע שהשתמשתי הוא postgresql ביצעתי את ההתקנה עפ"י
<https://www.youtube.com/watch?v=BLH3s5eTL4Y&t=440s> - הסרטון

יצרתי database בשם hmomember והתחברתי אליו דרך השרת בקובץ
 application.properties

```
spring.datasource.url=jdbc:postgresql://localhost:5432/hmomember
spring.datasource.username=postgres
spring.datasource.password=tick8587
spring.datasource.driver-class-name=org.postgresql.Driver
spring.jpa.hibernate.ddl-auto=update
spring.jpa.show-sql=true
spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.PostgreSQLDialect
spring.jpa.properties.hibernate.format_sql=true
server.error.include-message=always
spring.mvc.format.date=yyyy-MM-dd
server.port=8060
```

את הקוד כתבתי בשפת java עם שימוש ב-spring-boot -

Java Spring Boot הוא כלי קוד פתוח המקל על framework ליצירת
microservices ואפליקציית אינטרנט.

Postman - אתר שמתפקד כצד לקוח ונותן לנו אפשרות לבדוק את תקינות
הקוד.