

תרגיל מסכם צד לקוח - חלק ב' - מדעי המחשב - תשפ"ו

איך ה-AI משתלב באפליקציה שלכם?

בניגוד לפונקציות רגילות שכתבתם עד כה (כמו חישוב סכום הזמנה), כאן אנחנו משתמשים במודל שפה חכם (של OpenAI). התהליך עובד כך:

1. **איסוף נתונים:** אתם אוספים את היסטוריית ההזמנות של המשתמש (מה הוא קנה, מתי ובאיזה מחיר).
2. **שליחה לעיבוד (The Request):** הנתונים נשלחים ל-API. שם, ה-AI "קורא" את רשימת ההזמנות, מזהה דפוסים (למשל: "המשתמש אוהב סליטים בבוקר") ומגבש המלצות.
3. **קבלת תובנות (The Response):** ה-AI מחזיר לכם קטע קוד HTML מעוצב הכולל המלצות אישיות, כמו "נראה שאתה אוהב קפה חזק, כדאי לך לנסות את המקיאטו שלנו בחודש הבא!".

הנחיות לביצוע

כדי שהמערכת תעבוד בצורה חלקה, עליכם להקפיד על הדגשים הבאים:

- **מבנה הנתונים (JSON):** ה-AI מצפה לקבל מערך של אובייקטים.
- **שימוש ב-Fetch:** אתם משתמשים בשיטת `POST`. שימו לב שאתם הופכים את אובייקט הנתונים שלכם למחרוזת בעזרת `JSON.stringify`.
- **הצגת התוצאה:** ה-API מחזיר לכם HTML מוכן. אתם לא צריכים לעבד את הטקסט, אלא פשוט להזריק אותו לתוך אלמנט (כמו `div`) בדף שלכם בעזרת המאפיין `innerHTML`.

פורמט הנתונים שנשלח ל-API:

עליכם לשלוח בקשה בפורמט JSON הבקשה צריכה להכיל:

- עליכם לשלוח בקשה בפורמט JSON הבקשה צריכה להכיל:
- **transactions** מערך של אובייקטים, כשכל אובייקט מייצג טרנזקציה עם השדות: מערך של אובייקטים, כשכל אובייקט מייצג הזמנה עם השדות: `date`, `item`, `price`

לדוגמה:

```
{
  "transactions": [
    {
      "date": "2025-11-02",
      "item": "Espresso",
```

```
"price": 10.0
},
{
  "date": "2025-11-02",
  "item": "Croissant",
  "price": 12.0
},
{
  "date": "2025-11-05",
  "item": "Cappuccino",
  "price": 14.0
},
{
  "date": "2025-11-05",
  "item": "Greek Salad",
  "price": 45.0
},
{
  "date": "2025-11-10",
  "item": "Latte",
  "price": 15.0
},
{
  "date": "2025-11-10",
  "item": "Sandwich Tuna",
  "price": 38.0
},
{
  "date": "2025-12-01",
  "item": "Espresso",
  "price": 10.0
},
{
  "date": "2025-12-01",
  "item": "Chocolate Muffin",
  "price": 14.0
},
{
  "date": "2025-12-03",
  "item": "Cappuccino",
  "price": 14.0
},
{
```

```

    "date": "2025-12-03",
    "item": "Toast Cheese",
    "price": 28.0
  },
  {
    "date": "2025-12-08",
    "item": "Latte",
    "price": 15.0
  },
  {
    "date": "2025-12-08",
    "item": "Breakfast Combo",
    "price": 55.0
  }
]
}

```

השתמשו בפקודת fetch של JavaScript כדי לשלוח את הבקשה.

```

fetch('url',
{
  method: 'POST',
  body: JSON.stringify(/*HERE COMES YOUR ORDERS*/),
  headers: {
    'Content-type': 'application/json; charset=UTF-8',
  },
})
.then((response) => response.json())
.then((json) => /*HERE COMES THE RESPONSE*/);

```

לאחר שתקבלו את התגובה מה API הציגו את ההמלצות למשתמש (מוחזר לכם טקסט עם תגיות של HTML יש רק להציגו בדף)

להלן ה URL עבור הפניה ל API

<https://yael-ex-client-side-2026a-197579115104.me-west1.run.app/get-recommendations?api-key=afGre4Eerf223432AXE&lang=he>

להלן דוגמא לקריאת fetch ל API :

```
fetch(url, {
  method: 'POST', // Since we are sending data, use POST method
  headers: {
    'Content-Type': 'application/json', // Content type is JSON
  },
  body: JSON.stringify({ transactions }) // Send the transactions array in the body
})
.then(response => {
  if (!response.ok) {
    throw new Error(`HTTP error! Status: ${response.status}`);
  }
  return response.json();
}).then(data => {
  console.log("Recommendations:", data);
  // כאן תכתבו את הקוד להצגת ההמלצות בדף
}).catch(error => {
  console.error("Error fetching user information:", error);
});
```

אופן ההגשה

יש להעלות לתיקיה 6 - כלומר זו הגשה נפרדת (דומה להגשות עד כה)

יש לשלוח קישור ישיר לדף הבית

יש לשלוח קישור ישיר לדף ההמלצות (יש להוסיף באתר קישור לדף המלצות)

אם יש דף לוגין - יש לשלוח באותו מייל של הלינקים את שם המשתמש והסיסמא לכניסה

יש לשלוח קישור לזיפ - המכיל את כל הפרויקט של חלק א

בהצלחה!