Project Name:
'Ami Coding Pari Na'

Project Description:

You have to develop a web application. Your project will contain 3 sections.

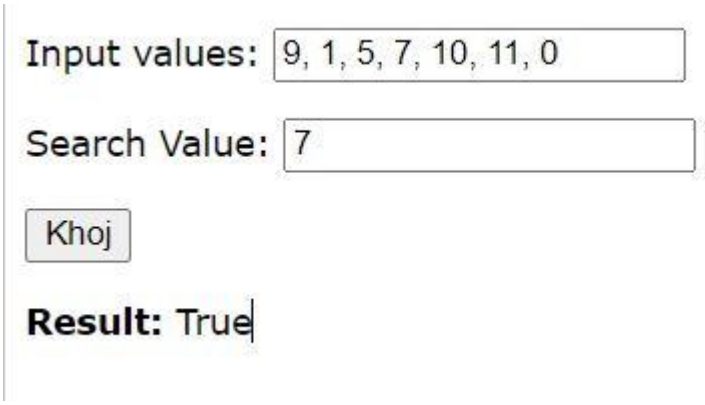
Section 1: User Authentication/Registration Page
A user login and registration section. You can use whatever input fields you want (maintaining a standard)

Section 2: Khoj the search Page
After login, users can access this page.

1. Khoj the search: In this segment(page), there will be two input fields

- Input Values: User can input comma separated integers
- Search Value: User can input only one integer
- Output: Will print **True** if the search value is in the input values. Otherwise print **False**

Sample



Now, before showing the output, you have to **store the input values in the database in sorted order(descending)** along with the logged in user id and the input timestamp. That means, when the user press the button "**Khoj**", the Input values (9, 1, 5, 7, 10, 11, 0) will be stored in the database as follows : 11, 10, 9, 7, 5, 1, 0

So the rough workflow for this section is as follows

1. Take the "Input Values"
2. Take the "Search Value"
3. Sort the "Input values" in descending order.
4. Store the sorted "Input Values" in the database.
5. Check if the "Search Value" is in the "Input Values"
6. Print the output

Note: The above workflow might not be the optimal workflow. You can change your workflow as you need to make it more optimized.

Section 3: API Endpoints

In this section, there will be only one API endpoints

Endpoint 1: Get All Input Values

Parameters: start_datetime, end_datetime, user_id

Returns: All the **Input Values** the user ever entered within start_datetime(inclusive) and end_datetime (inclusive). Check the following response format.

```
{
   “status”: “succes”,
   “user_id” : 1,
   “payload” : [
       {
           “timestamp” : ”2012-01-01 00:00:00”,
           “input_values” : “11, 10, 9, 7, 5, 1, 0”
        },
       {
             “timestamp” : ”2013-01-01 01:00:00”,
              “input_values” : “13, 11, 10, 7, 5, 2, 1”

     ]
}
```

Project Constraints:

1. You can use any framework you want to develop this application.
2. Think about programming complexity and design patterns so that your application can handle so many requests efficiently.
3. Provide proper commenting in your code.
4. Follow good coding practices.
5. You can use any CSS framework.

Submission Guideline:

1. Provide a **Readme** file containing detailed and specific instructions on how we can run your code.
2. Upload your code to a git repository.
3. Email your git link to: **career@evidentbd.com**

Bonus:

1. Token based authentication system in API
2. Dockerizing the application
3. Host the application in your own hosting
4. Creative way to design the frontend.