Full Stack Question Paper – VIT

Objective

Build and host a REST API (Method: POST) that takes in an array and returns the following:

1. Status
2. User ID
3. Email ID
4. College Roll Number
5. Array for even numbers
6. Array for odd numbers
7. Array for alphabets, converted to uppercase
8. Array for special characters
9. Sum of numbers
10. Concatenation of all alphabetical characters present in the input in the reverse order in alternating caps

Preferred Tech Stack

Node.js / Python / Java

Hosting

Any provider of your choice. If you don't have one already, please use Vercel / Railway / Render any other provider which supports REST API hosting.

Also push your code to a public Github Repository

Logic

Response should always contain "user_id" in the following format :
"user_id": {full_name_ddmmyyyy}
E.g.: "user_id": "john_doe_17091999"
NOTE: Full name must be in lowercase
"is_success" should be returned in the response to mark the status of operation. It can be true / false.
Be sure to follow other best practices and handle exceptions gracefully

Hosted API

Method: POST
Route: /bfhl
Expected status code for successful requests: 200
Example: https://testbfhl.herokuapp.com/bfhl

Submission

- Develop the API in any of the languages mentioned.
- Host your API (refer to III)
- Share the API endpoint in the form given to you (with "/bfhl")
- Form Link: https://forms.office.com/r/ZeVpUYp3zV
- Fastest valid submissions will be considered
- **IMPORTANT:**
    - The URL that you submit should have the logic specified above in “/bfhl” route
    - Numbers must also be returned as strings. For example, if "1" is present in the input data, the response must also contain 1 as a string ("1").

Examples

Example A

Request

```
{
"data": [“a”,”1”,”334”,”4”,”R”, “$”]
}
```

Response

```
{
"is_success": true,
"user_id": "john_doe_17091999",
“email” : “john@xyz.com”,
“roll_number”:”ABCD123”,
"odd_numbers": [“1”],
"even_numbers": [“334”,”4”],
"alphabets": [“A”,”R”],
“special_characters”: [“$”],
"sum": "339" // return sum as a string
"concat_string": "Ra"
}
```

Example B

Request

```
{
"data": [“2”,"a", "y", ”4”, ”&”, “-”, “*”,  ”5”,”92”,”b”]
}
```

Response

```
{
"is_success": true,
"user_id": "john_doe_17091999",
“email” : “john@xyz.com”,
“roll_number”:”ABCD123”,
"odd_numbers": [”5”],
"even_numbers": [“2”,”4”,”92”],
"alphabets": [“A”, "Y", “B”],
“sepcial_characters”: [“&”, “-”, “*”],
"sum": "103",
"concat_string": "ByA"
}
```

Example C

Request

```
{
"data": ["A",”ABcD”,”DOE”]
}
```

Response

```
{
"is_success": true,
"user_id": "john_doe_17091999",
“email” : “john@xyz.com”,
“roll_number”:”ABCD123”,
"odd_numbers": [],
"even_numbers": [],
"alphabets": ["A",”ABCD”,”DOE”],
“special_characters”: [],
"sum": "0",
"concat_string": "EoDdCbAa"
}
```