

TP-05

JavaScript

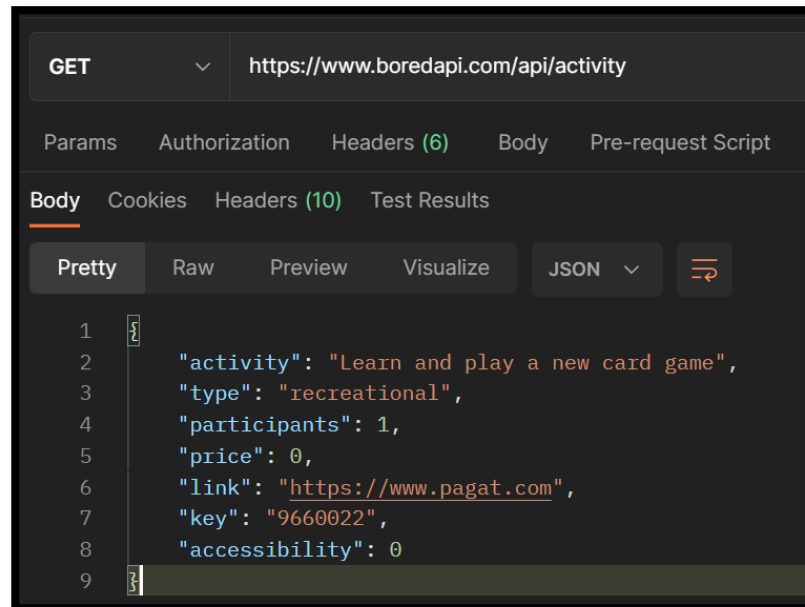
(Promise, Fetch)

TP05 Exercise

TP05.1: Activity Suggestion

Create a web app to random activity for suggesting activity to a boring person. Use **Fetch** function in JavaScript to request the data from the server.

You can access free API from: <https://www.boredapi.com/api/activity>



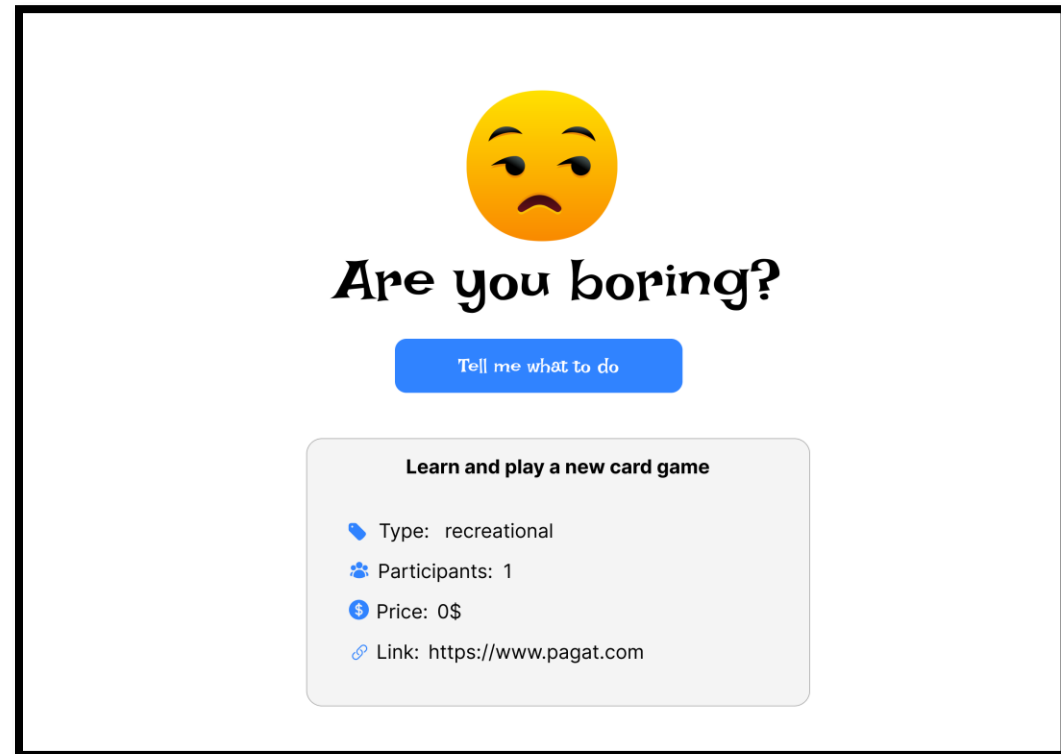
```
GET https://www.boredapi.com/api/activity

Params  Authorization  Headers (6)  Body  Pre-request Script

Body  Cookies  Headers (10)  Test Results

Pretty  Raw  Preview  Visualize  JSON  ↺

1  {
2    "activity": "Learn and play a new card game",
3    "type": "recreational",
4    "participants": 1,
5    "price": 0,
6    "link": "https://www.pagat.com",
7    "key": "9660022",
8    "accessibility": 0
9  }
```



TP05 Exercise

TP05.2: Gender Guessing

Create a web app to guess a person's gender by providing a person's name. Use **Fetch** function in JavaScript to request the data from the server.

You can access free API from: <https://api.genderize.io?name=Makara>



The screenshot shows a web browser's developer console with the following details:

- Method: GET
- URL: `https://api.genderize.io?name=Makara`
- Params: (none)
- Authorization: (none)
- Headers: (6)
- Body: (13)
- Test Results: (none)

The response body is displayed in JSON format:

```
1 {
2   "count": 691,
3   "gender": "female",
4   "name": "Makara",
5   "probability": 0.9
6 }
```



TP05 Exercise

TP05.3: Nationality Guessing

Create a web app to guess a person's nationality by providing a person's name. Use **Fetch** function in JavaScript to request the data from the server.

You can access free API from: <https://api.nationalize.io?name=makara>




```
GET https://api.nationalize.io?name=makara

Params Authorization Headers (6) Body Pre-request Script

Body Cookies Headers (13) Test Results

Pretty Raw Preview Visualize JSON ↗

1  [
2    "country": [
3      {
4        "country_id": "KH",
5        "probability": 0.81
6      },
7      {
8        "country_id": "LS",
9        "probability": 0.043
10     },
11     {
12       "country_id": "VN",
13       "probability": 0.018
14     },
15     {
16       "country_id": "TH",
17       "probability": 0.016
18     },
19     {
20       "country_id": "HU",
21       "probability": 0.013
22     }
23   ],
24   "name": "makara"
25 ]
```



I Know you Home

Guess

KH	81%
LS	4.3%
VN	1.8%
TH	1.6%
HU	1.3%

TP05 Exercise

TP05.4: University List

Create a web app to display a list of university of a country. Use **Fetch** function in JavaScript to request the data from the server.

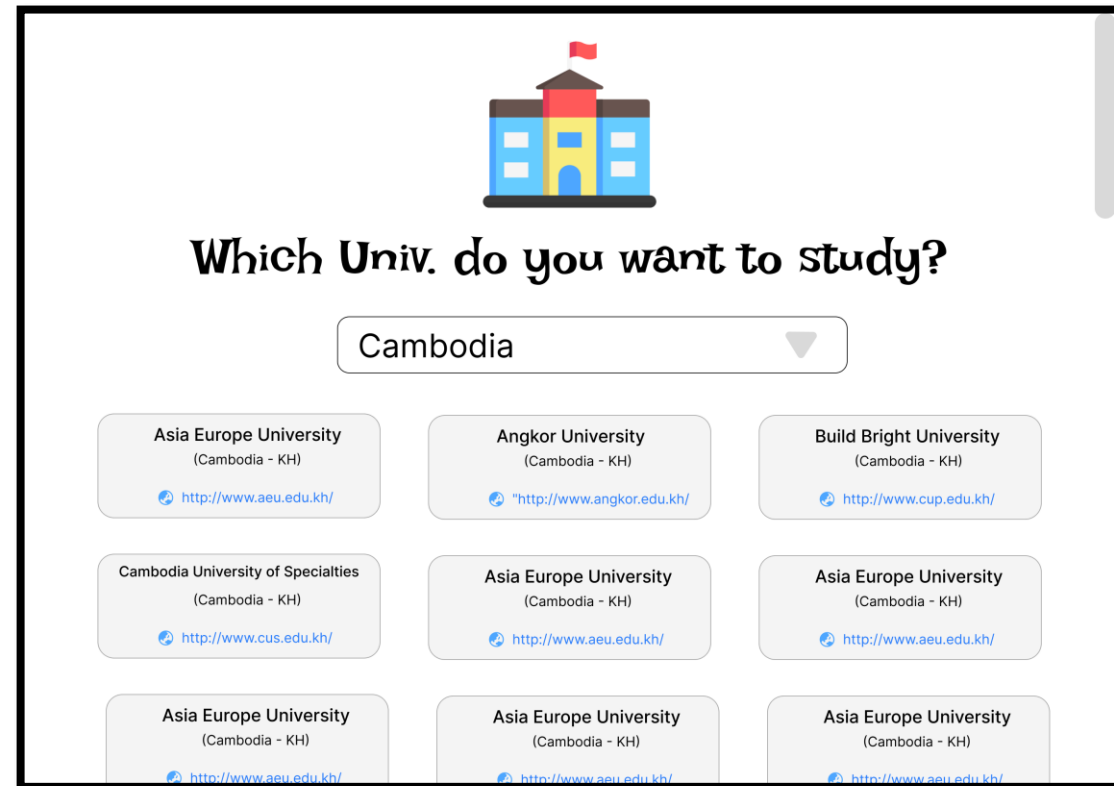
You can access free API from: <http://universities.hipolabs.com/search?country=Cambodia>



```
GET http://universities.hipolabs.com/search?country=cambodia

Body
Pretty Raw Preview Visualize JSON

1 [
2   {
3     "domains": [
4       "aeu.edu.kh"
5     ],
6     "country": "Cambodia",
7     "alpha_two_code": "KH",
8     "web_pages": [
9       "http://www.aeu.edu.kh/"
10    ],
11    "state-province": null,
12    "name": "Asia Europe University"
13  },
14  {
15    "domains": [
16      "angkor.edu.kh"
17    ],
18    "country": "Cambodia",
19    "alpha_two_code": "KH",
20    "web_pages": [
21      "http://www.angkor.edu.kh/"
22    ],
23    "state-province": null,
24    "name": "Angkor University"
25  },
26  }
```



Fetch data from API for boooook App

TP05.5: Book Store

Apply fetch function to our previous book app. Free Api from jsonplaceholder.typicode.com

■ All Photos Request

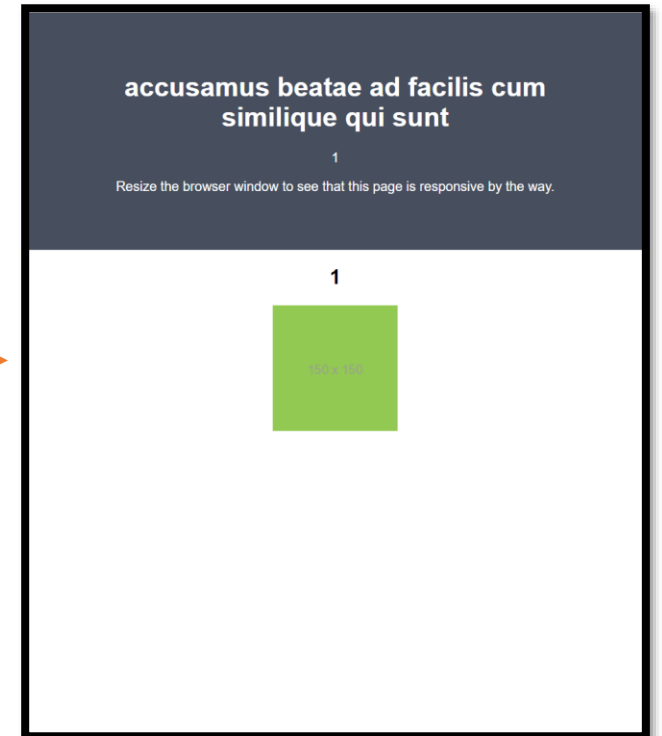
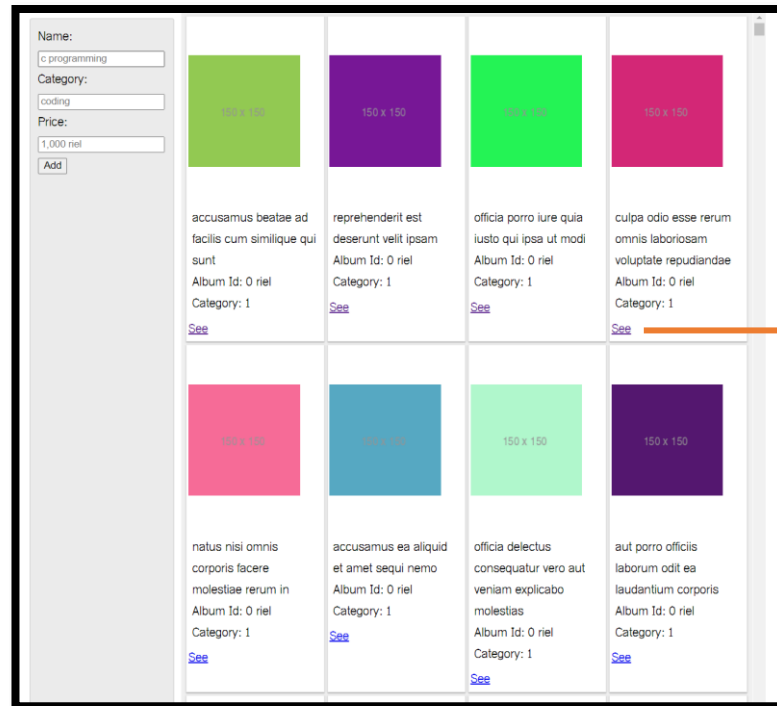
```
fetch('https://jsonplaceholder.typicode.com/photos')
  .then((response) => {
    return response.json();
  })
  .then((myJson) => {
    console.log(myJson);
  });
```

```
// {
//   albumId: 1
//   id: 1
//   thumbnailUrl: "https://via.placeholder.com/150/92c952"
//   title: "accusamus beatae ad facilis cum similique qui sunt"
//   url: "https://via.placeholder.com/600/92c952"
// }
```

■ Detail Info Request

```
fetch(`https://jsonplaceholder.typicode.com/photos/${bookId}`)
  .then((response) => {
    return response.json();
  })
  .then((myJson) => {
    console.log(myJson);
  });
```

```
{
  "albumId": 1,
  "id": 1,
  "title": "accusamus beatae ad facilis cum similique qui sunt",
  "url": "https://via.placeholder.com/600/92c952",
  "thumbnailUrl": "https://via.placeholder.com/150/92c952"
}
```





POSTMAN

Download

Getting to understand

“Promise” & “Fetch”

Good luck 🍀

<https://apiheny.io/free-api/#apis-without-key>