

Tkinter Web API - Create Your Own App

Contents

Tkinter Web API - Create Your Own App.....	1
Pseudocode	1
Here's What I Want You to Do	1
Here's Why I Want You to Do It	2
Program Minimum Requirements.....	2
Free Public Web API's	2
Program Examples	2
Joke API.....	3
Bored API.....	4
Assignment Submission.....	4

Time required: 90 minutes

Please read the directions carefully before beginning the assignment.

- Comment each line of code as shown in the tutorials and other code examples.
- Follow all directions carefully and accurately.
- Think of the directions as minimum requirements.

Pseudocode

1. Write pseudocode for the exercise
2. Save it in a document
3. Submit with the assignment

Here's What I Want You to Do

Earlier, we created a command line version of a program that displayed random jokes from a Free Public Web API. We are going to create a Tkinter GUI version.

Your program can display any type of information from a Web API.

Here's Why I Want You to Do It

Demonstrate understanding of:

Tkinter, OOP, requests, dictionary, API

Program Minimum Requirements

Use OOP.

1. Create a Python Tkinter program named for your project.
2. Display information from an API. Jokes, Cat Facts, Sports, etc.
3. When the user clicks the Get button, display the information.
4. The design is up to you. Be creative!

Free Public Web API's

These are only some of the free API resources available.

- <https://any-api.com>
- <https://apilist.fun>
- <https://github.com/public-apis/public-apis>
- <https://mixedanalytics.com/blog/list-actually-free-open-no-auth-needed-apis>
- <https://public-apis.io>
- jsonplaceholder.typicode.com/ (Test API with fake JSON data)

Program Examples

Your interface does not have to look like the examples. These are here to give you an example of how many Free Public API's there are and how they might be displayed.

Some API's are well documented, some are not. Do a Google search, maybe someone else has figured it out, or there might be a tutorial that can help.

It is a good idea to build a console version first. It is easier to build and troubleshoot.

To display the data in these API's, print out the dictionary. Some dictionaries are dictionaries of lists, sometimes they are nested dictionaries and list. Each one is a bit different.

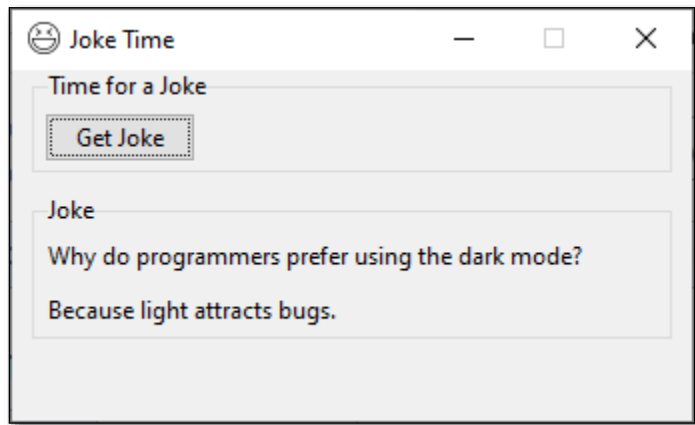
Joke API

```
# API URL for a single random safe joke
URL = 'https://v2.jokeapi.dev/joke/Any?type=twopart&safe-mode'
```

JSON

Take a look at the items in the dictionary we want. You will want to get setup and delivery.

```
{
  "error": false,
  "category": "Pun",
  "type": "twopart",
  "setup": "Where do sick cruise ships go to get healthy?",
  "delivery": "The dock!",
  "flags": {
    "nsfw": false,
    "religious": false,
    "political": true,
    "racist": false,
    "sexist": false,
    "explicit": false
  },
  "id": 212,
  "safe": true,
  "lang": "en"
}
```



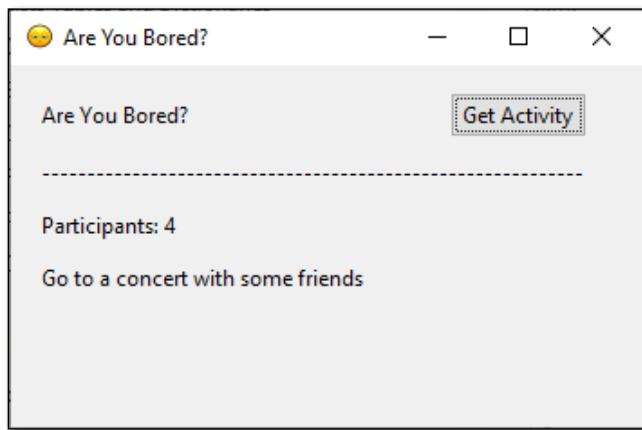
Bored API

```
# API URL for a random activity to do when you are bored
URL = 'https://bored-api.appbrewery.com/'
```

JSON

Take a look at the items in the dictionary we want. You will want to get **participants** and **activity**.

```
{
  "activity": "Bake a pie with some friends",
  "type": "cooking",
  "participants": 3,
  "price": 0.3,
  "link": "",
  "key": "3141592",
  "accessibility": 0.3
}
```



Assignment Submission

1. Attach the pseudocode and program
2. Insert a screenshot of the program running
3. Submit in Blackboard.