RepositoryApp Documention

The module repository_app includes the necessary tools to get the requested data. This document illustrates how to use the class and test it. The class uses the `request` package to get the information from the GitHub API.

First, we import the class:

In [1]:
```
from repository_app import RepositoryApp
```

We instantiate the class. It has as `url` as *public* instance attribute. `url` allows the user to define the source of information (by defect, `url` is "https://api.github.com").

In [2]:
```
app = RepositoryApp()
```

The class has three main methods to retrieve the information and calculate the requested answers:

Given an organization return the number of repositories:

The method `self.request_number_repositories` returns the solution to this question. The user provides the name of an organization, and it will compute the number of public repositories. The argument `org_name` is where the user can define the organization name:

In [3]:
```
app.request_number_repositories(org_name='GitHub')
```

```
Number public repositories: 409
```

Given an organization return the biggest repository (in bytes):

The method `self.request_biggest_repository` returns the size of the biggest repository. The user provides the name of an organization, and it will compute its size (in bytes). In addition, the function will return the name of the repository. The argument `org_name` is where the user can define the organization name:

In [4]:
```
app.request_biggest_repository(org_name='GitHub')
```

```
Size biggest repository: 151192576 bytes
Name repository: developer.github.com
```

Return the number of organizations that are currently on Github:

The method `self.request_number_organization` returns the number of current organizations in GitHub. It retrieves the information from the source `search/users` (`type=org`).

In [5]:
```
app.request_number_organization()
```

```
Number organizations on Github: 4453943
```