**Τελική Άσκηση – Δημιουργία ενός Github Repository μέσω API**

**Περιγραφή της Άσκησης :**

Στη συγκεκριμένη άσκηση καλούμαστε να δημιουργήσουμε ένα repository στο Github μέσω των APIs που διατίθενται στο χώρο για developers. Το repository θα έχει την ονομασία BIOS2020 και θα περιέχει δυο word files. Το παρόν που περιέχει όλη τη μεθοδολογία και ένα άλλο που θα περιέχει το link του repository. Το χτίσιμο του request γίνεται με το εργαλείο Postman που εγκαταστήσαμε στην τελευταία διάλεξη.

**ΒΗΜΑ 1 : API developer page**

* **Github -** <https://developer.github.com/>
* **Χώρος του συγκεκριμένου API :** <https://developer.github.com/v3/repos/#create>
* **Developer ID: kyrlef**
* **REST API v3 : Create**
* **What does the API do?**

Creates a new repository for the authenticated user.

* **Resource URL:** <https://api.github.com/user/repos>
* **Needs authorization**
* **Response formats :** JSON

**ΒΗΜΑ 2 : API analysis**

a) Δημιουργία public repository με όνομα BIOS2020 στο user endpoint.

b) API Template

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| name | string | **Required**. The name of the repository. |
| description | string | A short description of the repository. |
| homepage | string | A URL with more information about the repository. |
| private | boolean | Either true to create a private repository or false to create a public one. Creating private repositories requires a paid GitHub account. Default: false |
| visibility | string | Can be public or private. If your organization is associated with an enterprise account using GitHub Enterprise Cloud, visibility can also be internal. For more information, see "[Creating an internal repository](https://help.github.com/github/creating-cloning-and-archiving-repositories/creating-an-internal-repository)" in the GitHub Help documentation. The visibility parameter overrides the private parameter when you use both parameters with the nebula-preview preview header. |
| has\_issues | boolean | Either true to enable issues for this repository or false to disable them. Default: true. |
| has\_projects | boolean | Either true to enable projects for this repository or false to disable them. Default: true. **Note:** If you're creating a repository in an organization that has disabled repository projects, the default is false, and if you pass true, the API returns an error. |
| has\_wiki | boolean | Either true to enable the wiki for this repository or false to disable it. Default: true. |
| is\_template | boolean | Either true to make this repo available as a template repository or false to prevent it. Default: false. |
| team\_id | integer | The id of the team that will be granted access to this repository. This is only valid when creating a repository in an organization. |
| auto\_init | boolean | Pass true to create an initial commit with empty README. Default: false. |
| gitignore\_template | string | Desired language or platform [.gitignore template](https://github.com/github/gitignore) to apply. Use the name of the template without the extension. For example, "Haskell". |
| license\_template | string | Choose an [open source license template](https://choosealicense.com/) that best suits your needs, and then use the [license keyword](https://help.github.com/articles/licensing-a-repository/#searching-github-by-license-type) as the license\_template string. For example, "mit" or "mpl-2.0". |
| allow\_squash\_merge | boolean | Either true to allow squash-merging pull requests, or false to prevent squash-merging. Default: true |
| allow\_merge\_commit | boolean | Either true to allow merging pull requests with a merge commit, or false to prevent merging pull requests with merge commits. Default: true |
| allow\_rebase\_merge | boolean | Either true to allow rebase-merging pull requests, or false to prevent rebase-merging. Default: true. |
| delete\_branch\_on\_merge | boolean | Either true to allow automatically deleting head branches when pull requests are merged, or false to prevent automatic deletion. Default: false. |

c) Generic part - Specific part

|  |  |  |
| --- | --- | --- |
| API Generic Part | | |
| no | Component Name : | Description |
| 1 | API location : <https://api.github.com/user/repos> | Where the API can be found |

|  |  |  |  |
| --- | --- | --- | --- |
| API Specific Part | | | |
| **Νο** | **Name** | **Default** | **Description** |
| 1 | name=”BIOS2020” |  | The name of the repository |
| 2 | description=”This is a repository for the needs of the lesson Epixeirisiakes efarmoges se H/Y” |  | The description of the created repository |
| 3 | homepage=”https://tinyurl.com/y3bahcxx” |  | The with the description of the exercise |
| 4 | private=false | false | This is a public repository, not private |
| 5 | has\_issues=true |  | We enable issues for this repository |
| 6 | has\_projects=true |  | We enable projects for this repository. |
| 7 | has\_wiki=true |  | We enable wikis for this repository |

**ΒΗΜΑ 3 : Βλέπουμε το παράδειγμα**

1. **To JSON του Request**

{

"name": "Hello-World",

"description": "This is your first repository",

"homepage": "https://github.com",

"private": **false**,

"has\_issues": **true**,

"has\_projects": **true**,

"has\_wiki": **true**

}

1. **Το Json του Response**

Status: 201 Created

Location: https://api.github.com/repos/octocat/Hello-World

{

"id": 1296269,

"node\_id": "MDEwOlJlcG9zaXRvcnkxMjk2MjY5",

"name": "Hello-World",

"full\_name": "octocat/Hello-World",

"owner": {

"login": "octocat",

"id": 1,

"node\_id": "MDQ6VXNlcjE=",

"avatar\_url": "https://github.com/images/error/octocat\_happy.gif",

"gravatar\_id": "",

"url": "https://api.github.com/users/octocat",

"html\_url": "https://github.com/octocat",

"followers\_url": "https://api.github.com/users/octocat/followers",

"following\_url": "https://api.github.com/users/octocat/following{/other\_user}",

"gists\_url": "https://api.github.com/users/octocat/gists{/gist\_id}",

"starred\_url": "https://api.github.com/users/octocat/starred{/owner}{/repo}",

"subscriptions\_url": "https://api.github.com/users/octocat/subscriptions",

"organizations\_url": "https://api.github.com/users/octocat/orgs",

"repos\_url": "https://api.github.com/users/octocat/repos",

"events\_url": "https://api.github.com/users/octocat/events{/privacy}",

"received\_events\_url": "https://api.github.com/users/octocat/received\_events",

"type": "User",

"site\_admin": **false**

},

"private": **false**,

"html\_url": "https://github.com/octocat/Hello-World",

"description": "This your first repo!",

"fork": **false**,

"url": "https://api.github.com/repos/octocat/Hello-World",

"archive\_url": "http://api.github.com/repos/octocat/Hello-World/{archive\_format}{/ref}",

"assignees\_url": "http://api.github.com/repos/octocat/Hello-World/assignees{/user}",

"blobs\_url": "http://api.github.com/repos/octocat/Hello-World/git/blobs{/sha}",

"branches\_url": "http://api.github.com/repos/octocat/Hello-World/branches{/branch}",

"collaborators\_url": "http://api.github.com/repos/octocat/Hello-World/collaborators{/collaborator}",

"comments\_url": "http://api.github.com/repos/octocat/Hello-World/comments{/number}",

"commits\_url": "http://api.github.com/repos/octocat/Hello-World/commits{/sha}",

"compare\_url": "http://api.github.com/repos/octocat/Hello-World/compare/{base}...{head}",

"contents\_url": "http://api.github.com/repos/octocat/Hello-World/contents/{+path}",

"contributors\_url": "http://api.github.com/repos/octocat/Hello-World/contributors",

"deployments\_url": "http://api.github.com/repos/octocat/Hello-World/deployments",

"downloads\_url": "http://api.github.com/repos/octocat/Hello-World/downloads",

"events\_url": "http://api.github.com/repos/octocat/Hello-World/events",

"forks\_url": "http://api.github.com/repos/octocat/Hello-World/forks",

"git\_commits\_url": "http://api.github.com/repos/octocat/Hello-World/git/commits{/sha}",

"git\_refs\_url": "http://api.github.com/repos/octocat/Hello-World/git/refs{/sha}",

"git\_tags\_url": "http://api.github.com/repos/octocat/Hello-World/git/tags{/sha}",

"git\_url": "git:github.com/octocat/Hello-World.git",

"issue\_comment\_url": "http://api.github.com/repos/octocat/Hello-World/issues/comments{/number}",

"issue\_events\_url": "http://api.github.com/repos/octocat/Hello-World/issues/events{/number}",

"issues\_url": "http://api.github.com/repos/octocat/Hello-World/issues{/number}",

"keys\_url": "http://api.github.com/repos/octocat/Hello-World/keys{/key\_id}",

"labels\_url": "http://api.github.com/repos/octocat/Hello-World/labels{/name}",

"languages\_url": "http://api.github.com/repos/octocat/Hello-World/languages",

"merges\_url": "http://api.github.com/repos/octocat/Hello-World/merges",

"milestones\_url": "http://api.github.com/repos/octocat/Hello-World/milestones{/number}",

"notifications\_url": "http://api.github.com/repos/octocat/Hello-World/notifications{?since,all,participating}",

"pulls\_url": "http://api.github.com/repos/octocat/Hello-World/pulls{/number}",

"releases\_url": "http://api.github.com/repos/octocat/Hello-World/releases{/id}",

"ssh\_url": "git@github.com:octocat/Hello-World.git",

"stargazers\_url": "http://api.github.com/repos/octocat/Hello-World/stargazers",

"statuses\_url": "http://api.github.com/repos/octocat/Hello-World/statuses/{sha}",

"subscribers\_url": "http://api.github.com/repos/octocat/Hello-World/subscribers",

"subscription\_url": "http://api.github.com/repos/octocat/Hello-World/subscription",

"tags\_url": "http://api.github.com/repos/octocat/Hello-World/tags",

"teams\_url": "http://api.github.com/repos/octocat/Hello-World/teams",

"trees\_url": "http://api.github.com/repos/octocat/Hello-World/git/trees{/sha}",

"clone\_url": "https://github.com/octocat/Hello-World.git",

"mirror\_url": "git:git.example.com/octocat/Hello-World",

"hooks\_url": "http://api.github.com/repos/octocat/Hello-World/hooks",

"svn\_url": "https://svn.github.com/octocat/Hello-World",

"homepage": "https://github.com",

"language": **null**,

"forks\_count": 9,

"stargazers\_count": 80,

"watchers\_count": 80,

"size": 108,

"default\_branch": "master",

"open\_issues\_count": 0,

"is\_template": **true**,

"topics": [

"octocat",

"atom",

"electron",

"api"

],

"has\_issues": **true**,

"has\_projects": **true**,

"has\_wiki": **true**,

"has\_pages": **false**,

"has\_downloads": **true**,

"archived": **false**,

"disabled": **false**,

"visibility": "public",

"pushed\_at": "2011-01-26T19:06:43Z",

"created\_at": "2011-01-26T19:01:12Z",

"updated\_at": "2011-01-26T19:14:43Z",

"permissions": {

"admin": **false**,

"push": **false**,

"pull": **true**

},

"allow\_rebase\_merge": **true**,

"template\_repository": **null**,

"temp\_clone\_token": "ABTLWHOULUVAXGTRYU7OC2876QJ2O",

"allow\_squash\_merge": **true**,

"allow\_merge\_commit": **true**,

"subscribers\_count": 42,

"network\_count": 0

}

**ΒΗΜΑ 4 : Φτιάχνουμε το δικό μας API με τη βοήθεια του POSTMAN**

Πριν να μπορέσουμε να χτίσουμε το Request για το API πρέπει να ακολουθήσουμε τα εξής βήματα:

Δημιουργία Personal access token στο Github.

* Στο Settings του Github > Personal Settings > Developer settings επιλέγουμε Personal access tokens.
* Πατάμε Generate new token και κάνουμε copy to token που δημιουργήθηκε. Αυτό θα το τοποθετήσουμε στο Postman ώστε να μπορεί να κάνει authorization και να μας δίνει access στο API του Github.

Δημιουργία environment στο Postman

* Ανοίγουμε το Postman και στο workspace που επιθυμούμε κάνουμε new enivornment. Του δίνουμε το όνομα π.χ Github και ορίζουμε μια μεταβλητή token στην οποία εκχωρούμε το Personal access token που μας έδωσε το Github.

Δημιουργία του Request

* Επιλέγουμε POST και δίνουμε το URL του API : <https://api.github.com/user/repos>
* Επιλέγουμε Authorization > Bearer Token και στο πεδίο Token : {{token}} δηλαδή το όνομα της μεταβλητής που ορίσαμε στο environment.
* Επιλέγουμε body , raw , JSON και γράφουμε τον κώδικα :

{

"name": "BIOS2020",

"description": "This is a repository for the needs of the lesson Epixeirisiakes efarmoges se H/Y",

"homepage": "https://tinyurl.com/y3bahcxx",

"private": false,

"has\_issues": true,

"has\_projects": true,

"has\_wiki": true

}

* Πατάμε send και το repository δημιουργείται με επιτυχία.

1. **Request Message σε για HTTP**

POST /user/repos HTTP/1.1

Host: api.github.com

Content-Type: application/json

Authorization: Bearer 4b54bcbc768e3dde1d79c165148595f1c5fde1f8

{

"name": "BIOS2020",

"description": "This is a repository for the needs of the lesson Epixeirisiakes efarmoges se H/Y",

"homepage": "https://tinyurl.com/y3bahcxx",

"private": false,

"has\_issues": true,

"has\_projects": true,

"has\_wiki": true

}

**B) Expected Response Message**

{

    "id": 237671142,

    "node\_id": "MDEwOlJlcG9zaXRvcnkyMzc2NzExNDI=",

    "name": "BIOS2020",

    "full\_name": "kyrlef/BIOS2020",

    "private": false,

    "owner": {

        "login": "kyrlef",

        "id": 60289870,

        "node\_id": "MDQ6VXNlcjYwMjg5ODcw",

        "avatar\_url": "https://avatars1.githubusercontent.com/u/60289870?v=4",

        "gravatar\_id": "",

        "url": "https://api.github.com/users/kyrlef",

        "html\_url": "https://github.com/kyrlef",

        "followers\_url": "https://api.github.com/users/kyrlef/followers",

        "following\_url": "https://api.github.com/users/kyrlef/following{/other\_user}",

        "gists\_url": "https://api.github.com/users/kyrlef/gists{/gist\_id}",

        "starred\_url": "https://api.github.com/users/kyrlef/starred{/owner}{/repo}",

        "subscriptions\_url": "https://api.github.com/users/kyrlef/subscriptions",

        "organizations\_url": "https://api.github.com/users/kyrlef/orgs",

        "repos\_url": "https://api.github.com/users/kyrlef/repos",

        "events\_url": "https://api.github.com/users/kyrlef/events{/privacy}",

        "received\_events\_url": "https://api.github.com/users/kyrlef/received\_events",

        "type": "User",

        "site\_admin": false

    },

    "html\_url": "https://github.com/kyrlef/BIOS2020",

    "description": "This is a repository for the needs of the lesson Epixeirisiakes efarmoges se H/Y",

    "fork": false,

    "url": "https://api.github.com/repos/kyrlef/BIOS2020",

    "forks\_url": "https://api.github.com/repos/kyrlef/BIOS2020/forks",

    "keys\_url": "https://api.github.com/repos/kyrlef/BIOS2020/keys{/key\_id}",

    "collaborators\_url": "https://api.github.com/repos/kyrlef/BIOS2020/collaborators{/collaborator}",

    "teams\_url": "https://api.github.com/repos/kyrlef/BIOS2020/teams",

    "hooks\_url": "https://api.github.com/repos/kyrlef/BIOS2020/hooks",

    "issue\_events\_url": "https://api.github.com/repos/kyrlef/BIOS2020/issues/events{/number}",

    "events\_url": "https://api.github.com/repos/kyrlef/BIOS2020/events",

    "assignees\_url": "https://api.github.com/repos/kyrlef/BIOS2020/assignees{/user}",

    "branches\_url": "https://api.github.com/repos/kyrlef/BIOS2020/branches{/branch}",

    "tags\_url": "https://api.github.com/repos/kyrlef/BIOS2020/tags",

    "blobs\_url": "https://api.github.com/repos/kyrlef/BIOS2020/git/blobs{/sha}",

    "git\_tags\_url": "https://api.github.com/repos/kyrlef/BIOS2020/git/tags{/sha}",

    "git\_refs\_url": "https://api.github.com/repos/kyrlef/BIOS2020/git/refs{/sha}",

    "trees\_url": "https://api.github.com/repos/kyrlef/BIOS2020/git/trees{/sha}",

    "statuses\_url": "https://api.github.com/repos/kyrlef/BIOS2020/statuses/{sha}",

    "languages\_url": "https://api.github.com/repos/kyrlef/BIOS2020/languages",

    "stargazers\_url": "https://api.github.com/repos/kyrlef/BIOS2020/stargazers",

    "contributors\_url": "https://api.github.com/repos/kyrlef/BIOS2020/contributors",

    "subscribers\_url": "https://api.github.com/repos/kyrlef/BIOS2020/subscribers",

    "subscription\_url": "https://api.github.com/repos/kyrlef/BIOS2020/subscription",

    "commits\_url": "https://api.github.com/repos/kyrlef/BIOS2020/commits{/sha}",

    "git\_commits\_url": "https://api.github.com/repos/kyrlef/BIOS2020/git/commits{/sha}",

    "comments\_url": "https://api.github.com/repos/kyrlef/BIOS2020/comments{/number}",

    "issue\_comment\_url": "https://api.github.com/repos/kyrlef/BIOS2020/issues/comments{/number}",

    "contents\_url": "https://api.github.com/repos/kyrlef/BIOS2020/contents/{+path}",

    "compare\_url": "https://api.github.com/repos/kyrlef/BIOS2020/compare/{base}...{head}",

    "merges\_url": "https://api.github.com/repos/kyrlef/BIOS2020/merges",

    "archive\_url": "https://api.github.com/repos/kyrlef/BIOS2020/{archive\_format}{/ref}",

    "downloads\_url": "https://api.github.com/repos/kyrlef/BIOS2020/downloads",

    "issues\_url": "https://api.github.com/repos/kyrlef/BIOS2020/issues{/number}",

    "pulls\_url": "https://api.github.com/repos/kyrlef/BIOS2020/pulls{/number}",

    "milestones\_url": "https://api.github.com/repos/kyrlef/BIOS2020/milestones{/number}",

    "notifications\_url": "https://api.github.com/repos/kyrlef/BIOS2020/notifications{?since,all,participating}",

    "labels\_url": "https://api.github.com/repos/kyrlef/BIOS2020/labels{/name}",

    "releases\_url": "https://api.github.com/repos/kyrlef/BIOS2020/releases{/id}",

    "deployments\_url": "https://api.github.com/repos/kyrlef/BIOS2020/deployments",

    "created\_at": "2020-02-01T20:07:48Z",

    "updated\_at": "2020-02-01T20:07:48Z",

    "pushed\_at": "2020-02-01T20:07:49Z",

    "git\_url": "git://github.com/kyrlef/BIOS2020.git",

    "ssh\_url": "git@github.com:kyrlef/BIOS2020.git",

    "clone\_url": "https://github.com/kyrlef/BIOS2020.git",

    "svn\_url": "https://github.com/kyrlef/BIOS2020",

    "homepage": "https://tinyurl.com/y3bahcxx",

    "size": 0,

    "stargazers\_count": 0,

    "watchers\_count": 0,

    "language": null,

    "has\_issues": true,

    "has\_projects": true,

    "has\_downloads": true,

    "has\_wiki": true,

    "has\_pages": false,

    "forks\_count": 0,

    "mirror\_url": null,

    "archived": false,

    "disabled": false,

    "open\_issues\_count": 0,

    "license": null,

    "forks": 0,

    "open\_issues": 0,

    "watchers": 0,

    "default\_branch": "master",

    "permissions": {

        "admin": true,

        "push": true,

        "pull": true

    },

    "allow\_squash\_merge": true,

    "allow\_merge\_commit": true,

    "allow\_rebase\_merge": true,

    "delete\_branch\_on\_merge": false,

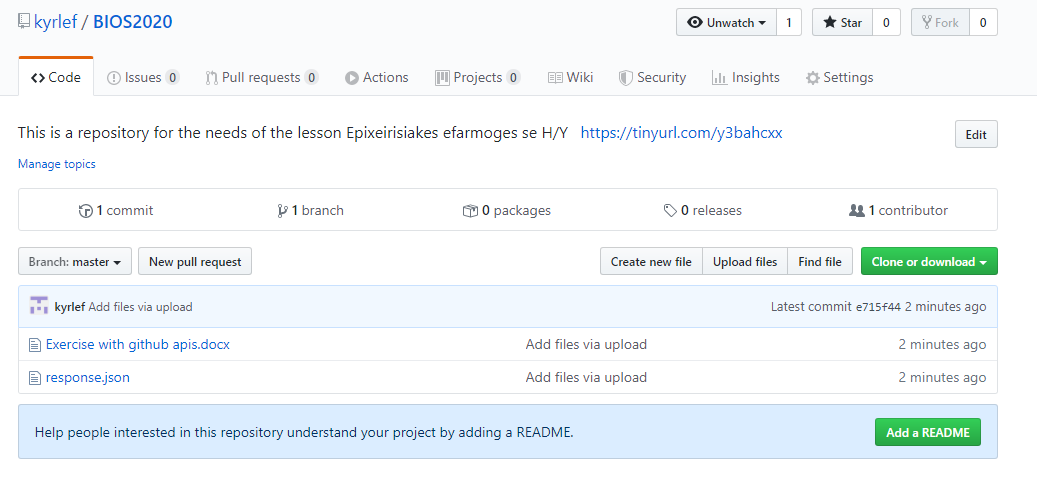
    "network\_count": 0,

    "subscribers\_count": 1

}

**ΒΗΜΑ 5 : Βλέπουμε το αποτέλεσμα**

**Το link του repository είναι :** https://github.com/kyrlef/BIOS2020.git

****