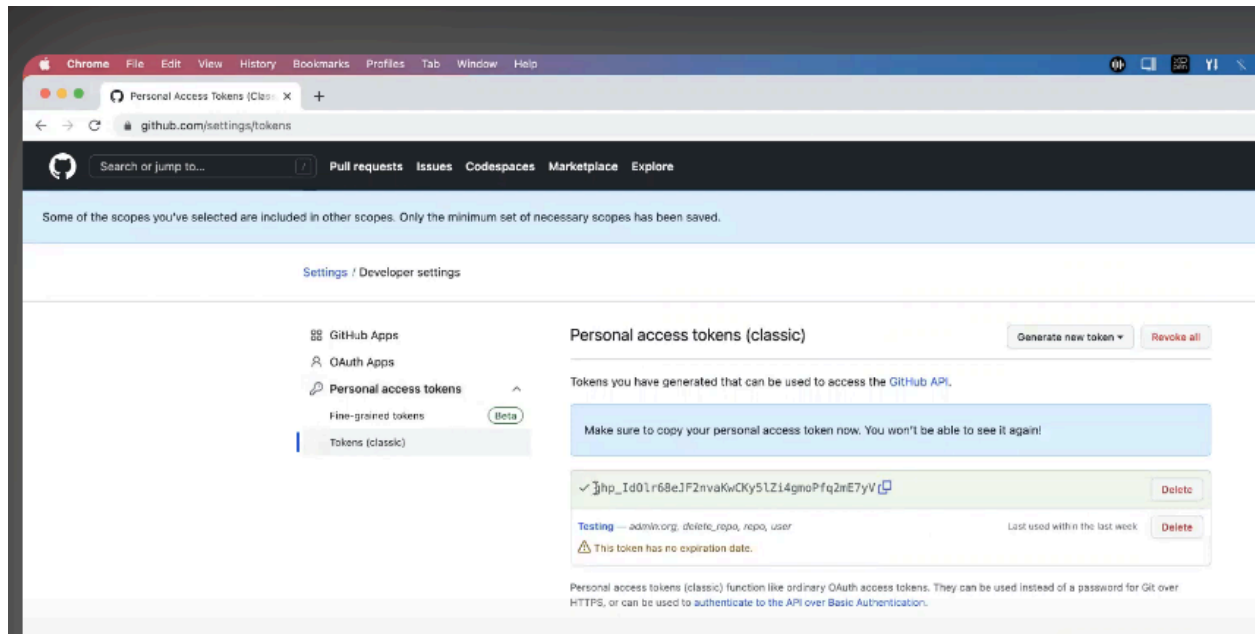


```
1 describe("Authentication",()=>{
2
3     it("Basic Authentication",()=>{
4         cy.request(
5             {
6                 method: 'GET',
7                 url: 'https://postman-echo.com/basic-auth',
8                 auth:{
9                     user: 'postman',
10                    pass:'password'
11                }
12            })
13        .then((response)=>{
14
15            expect(response.status).to.eq(200)
16            expect(response.body.authenticated).to.eq(true);
17
18        })
19    })
20
21    it("Digest Authentication",()=>{
22        cy.request(
23            {
24                method: 'GET',
25                url: 'https://postman-echo.com/basic-auth',
26                auth:{
27                    username: 'postman',
28                    password:'password',
29                    method: 'degest'
30                }
31            })
32        .then((response)=>{
33
34            expect(response.status).to.eq(200)
35            expect(response.body.authenticated).to.eq(true);
36
37        })
38    })
39 })
```

Github → click on profile – settings – in left side at bottom developer settings – expand personal access token – classic tokens.

From where to generate.

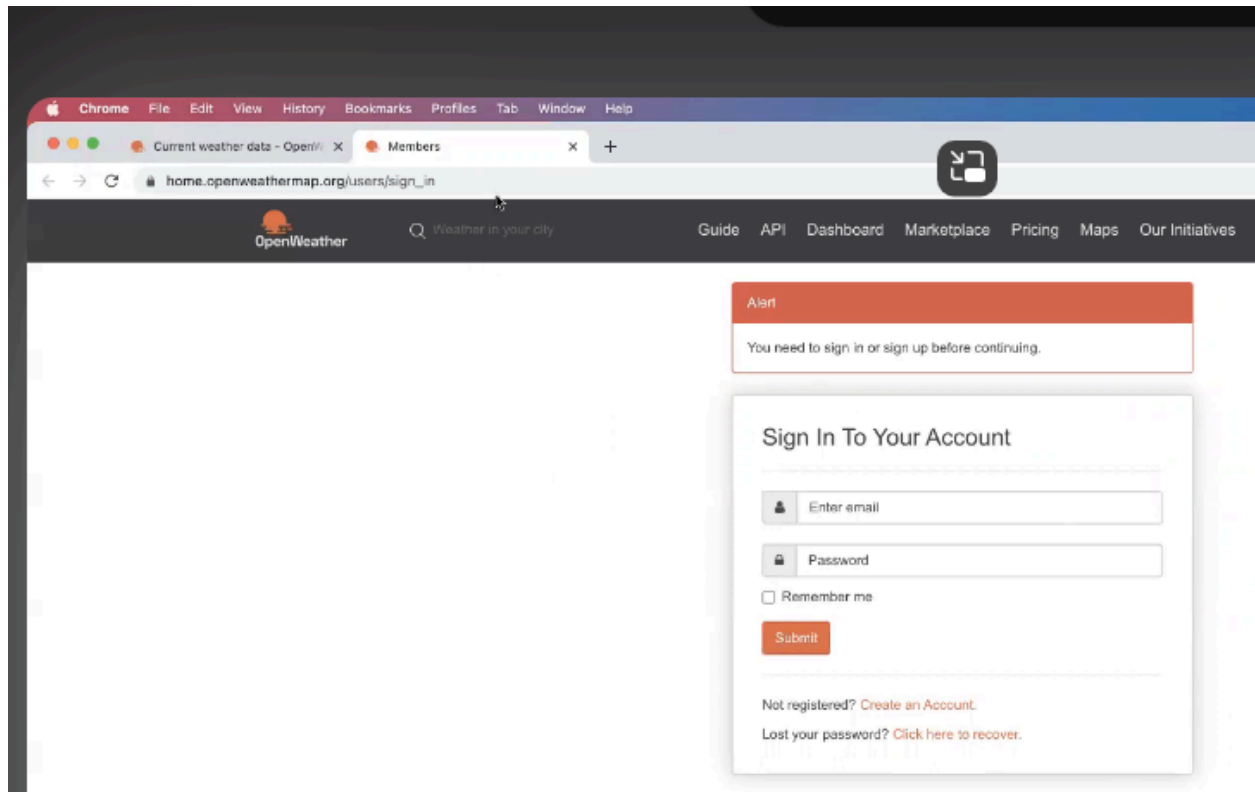
This is the same classic token we generate.



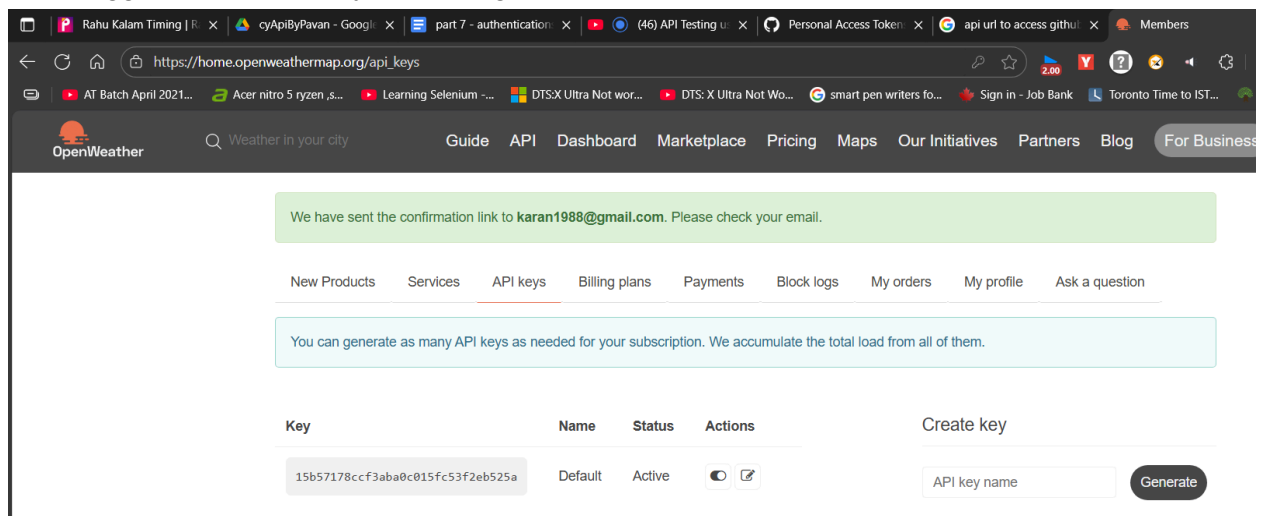
```
1 //this is nothing but token to access app.
2 //github personal access token generate and pass it here.
3 //it will give all your repos.
4
5     const token='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'
6     it("Bearer Token Auth",()=>{
7         cy.request(
8             {
9                 method: 'GET',
10                url: 'https://api.github.com/user/repos',
11                headers:{
12                    Authorization:'Bearer '+token
13                }
14            })
15        .then((response)=>{
16            expect(response.status).to.eq(200)
17        })
18    })
```

codesnap.dev

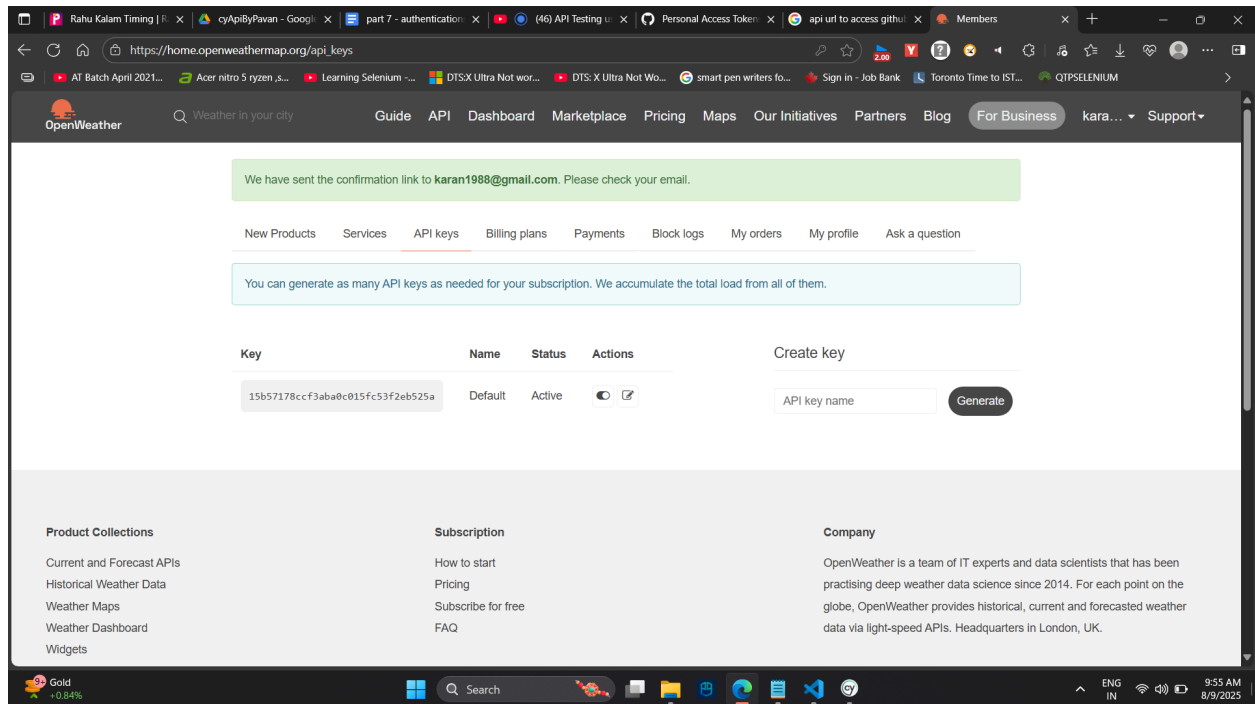
Go to weather api.  
Login.



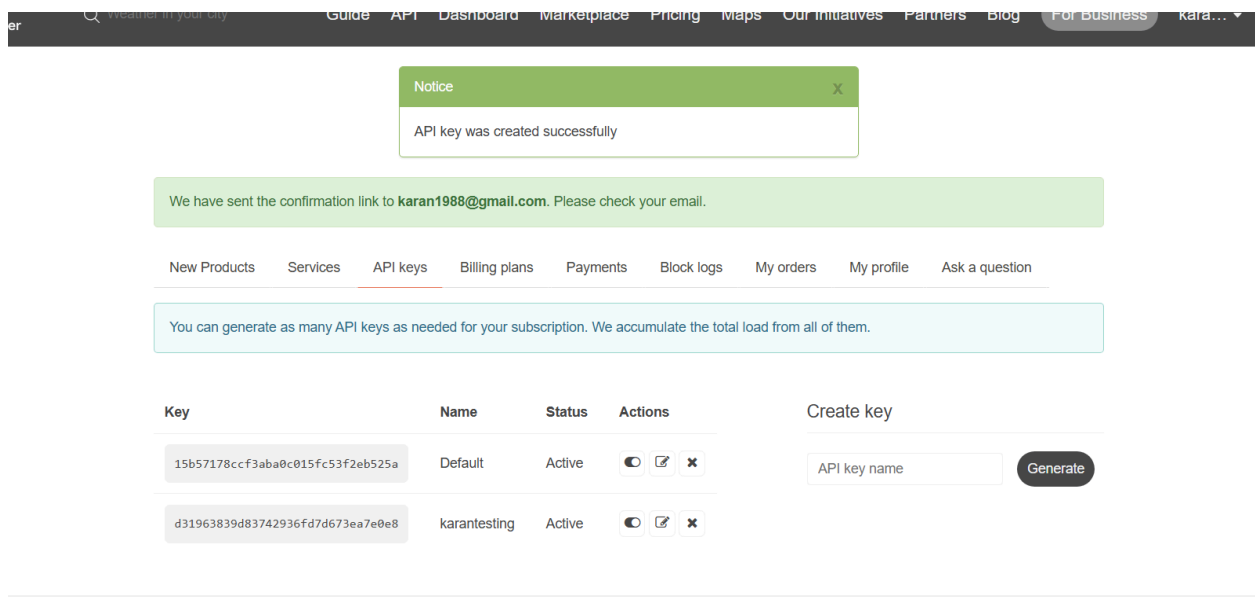
We should be able to generate api key.  
Once logged in click api keys and we get this-



Ocd



If you want write api key name and generate -



OpenWeather

Weather in your city

Guide API Dashboard Marketplace Pricing Maps Our Initiatives Partners Blog For Business







Notice

API key was created successfully

We have sent the confirmation link to **karan1988@gmail.com**. Please check your email.

New Products Services **API keys** Billing plans Payments Block logs My orders My profile Ask a question

You can generate as many API keys as needed for your subscription. We accumulate the total load from all of them.

Key	Name	Status	Actions
15b57178ccf3aba0c015fc53f2eb525a	Default	Active	  
ef0e9c0fa52ab81aac3f60d579e4f74d	karantesting	Active	  

Create key

API key name

We can use karantesting as api key name also –

```
openweatherapikey.cy.js x
cypress > e2e > openweatherapikey.cy.js > describe('api key for weather update') callback > it('open weath
1  /// <reference types="cypress" />
2
3  //go to the app, login, click api keys, generate one and use it .
4  //now we can access the app in our api calls.
5
6  describe('api key for weather update', () => {
7
8      it('open weather website', () => {
9
10         cy.request(/
```

```

9
10     cy.request({
11
12         method: 'get',
13         url: 'https://api.openweathermap.org/data/2.5/forecast/daily',
14         failOnStatusCode: false,
15         qs: {
16             q: 'delhi',
17             //we can use the name we used in weather api as the api key name.
18             //and the key we generate is the value.
19             karantesting: 'ef0e9c0fa52ab81aac3f60d579e4f74d'
20         }

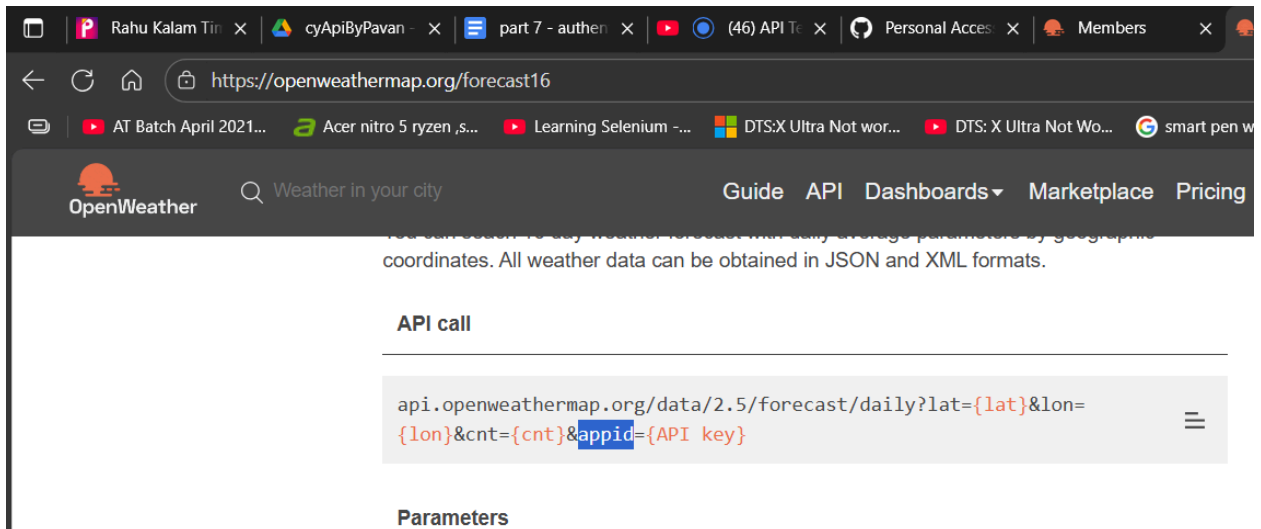
```

```

20     }
21 })
22 .then((response) => {
23     // expect(response.status).to.eq(200)
24     expect(response.status).to.eq(401)
25 })
26 });
27 });

```

We can also use appid as the key name as per the doc-



The screenshot shows the OpenWeather API documentation page. The browser address bar displays `https://openweathermap.org/forecast16`. The page header includes the OpenWeather logo, a search bar, and navigation links: Guide, API, Dashboards, Marketplace, and Pricing. The main content area is titled "API call" and shows the following API endpoint:

```
api.openweathermap.org/data/2.5/forecast/daily?lat={lat}&lon={lon}&cnt={cnt}&appid={API key}
```

Below the API call, the section "Parameters" is visible.

```
openweatherapikey.cy.js × openweatherapikey_1.cy.js ×
cypress > e2e > openweatherapikey_1.cy.js > describe("api key for weather update") callback > it("open weather website") callba
1  /// <reference types="cypress" />>
2
3  //go to the app, login, click api keys, generate one and use it .
4  //now we can access the app in our api calls.
5
6  describe("api key for weather update", () => {
7
8      it('open weather website', () => {
9
10         cy.request({
11
```

```
11
12         method: 'get',
13         url: 'https://api.openweathermap.org/data/2.5/forecast/daily',
14         failOnStatusCode: false,
15         qs: {
16             q: 'delhi',
17             //we can also use appid which is given in api doc.
18             //and the key we generate is the value.
19             appid: 'ef0e9c0fa52ab81aac3f60d579e4f74d'
20         }
21     })
22     .then((response)=>{
23         // expect(response.status).to.eq(200)
24         expect(response.status).to.eq(401)
25     })
```

```
25     })
26   });
27 });
```



```
1 //goto open weather website.
2 //current weather - api doc.
3 //sign in to weather app to generate api key.
4
5     it("API key Auth",()=>{
6
7         cy.request(
8             {
9                 method:"GET",
10                url:'api.openweathermap.org/data/2.5/forecast/daily',
11                qs:{
12                    q:'Delhi',
13                   appid:'fe9c5cddb7e01d747b4611c3fc9eaf2c' // API key and value
14                }
15            })
16        .then((response)=>{
17            expect(response.status).to.eq(200);
18        })
19
20    })
```

```
1 //we can also pass query parameter in this way.
2 //partially in url and partially in the qa.
3 it.only("API key Auth_1",()=>{
4
5     cy.request(
6         {
7             method:"GET",
8             url:'api.openweathermap.org/data/2.5/forecast/daily?q=Delhi',
9             qs:{
10                 appid:'fe9c5cddb7e01d747b4611c3fc9eaf2c' // API key and value
11             }
12         })
13     .then((response)=>{
14         expect(response.status).to.eq(200);
15     })
16
17
18
19 })
20
21 })
```