

Privacy and Security in Online Social Networks
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Lecture – 07
Tutorial 2 Part 1 Facebook API

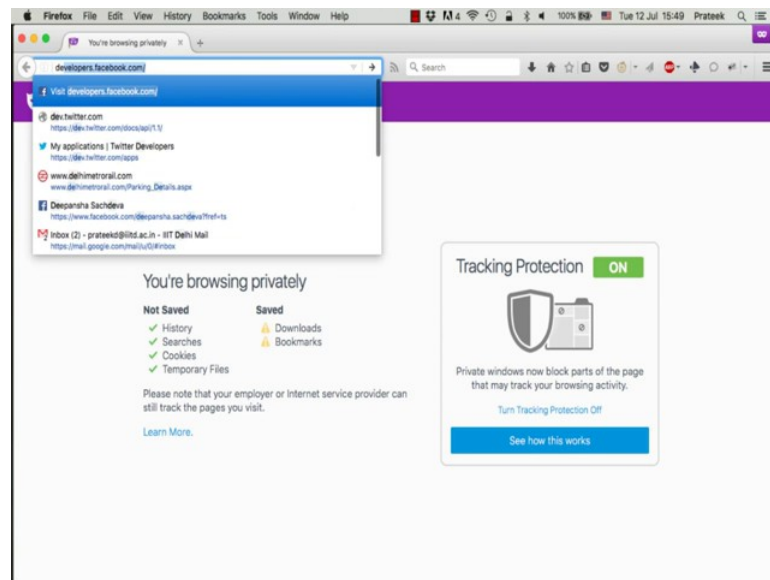
Hi everyone, welcome to the second tutorial for the **PSOSM** course.

(Refer Slide Time: 00:12)



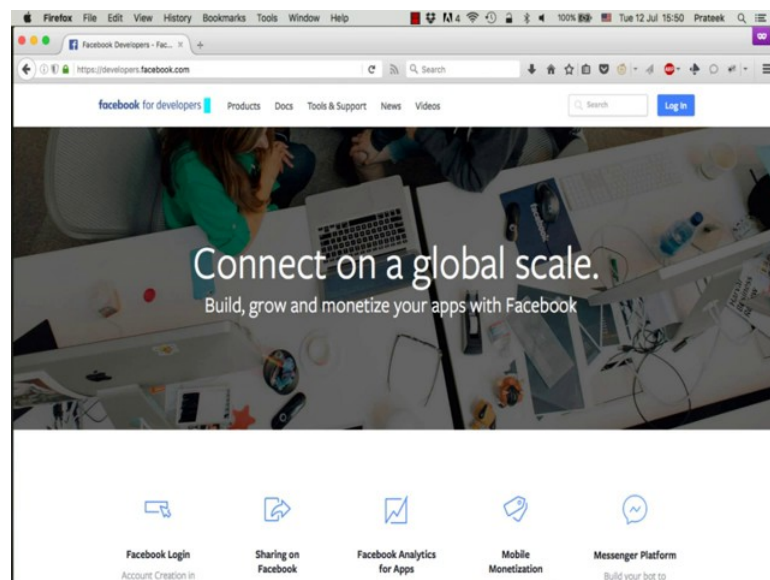
In the previous tutorial, we learnt how to install ubuntu and got started with python. We also saw some basics about package management in ubuntu and python. Now in this tutorial, we will learn how to collect data from Facebook using the graph API. You must have browsed Facebook as one of the millions of common Facebook users hundreds of times, but now we will introduce Facebook to you from a developer's perspective which is entirely different from what you **must have** seen until now.

(Refer Slide Time: 00:45)



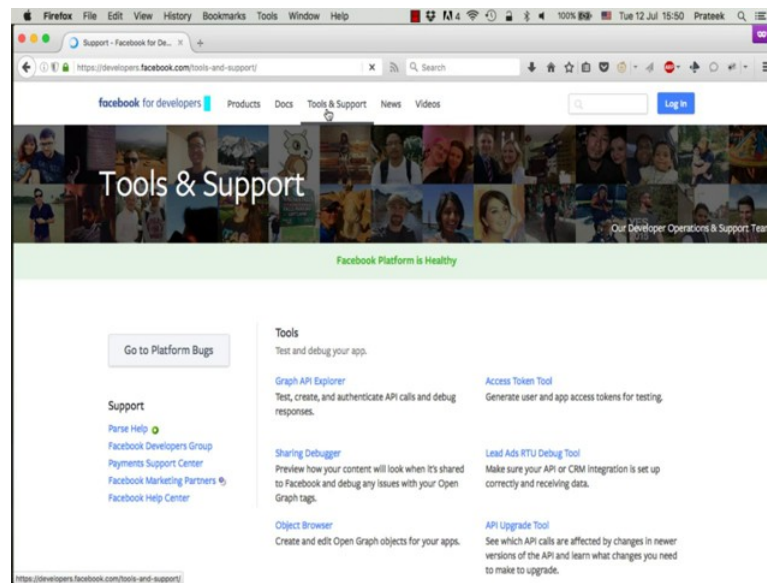
To start open your browser and open developers dot facebook dot com. We will see a glimpse of what Facebook has in **store** for the developer's side of **you**.

(Refer Slide Time: 00:55)

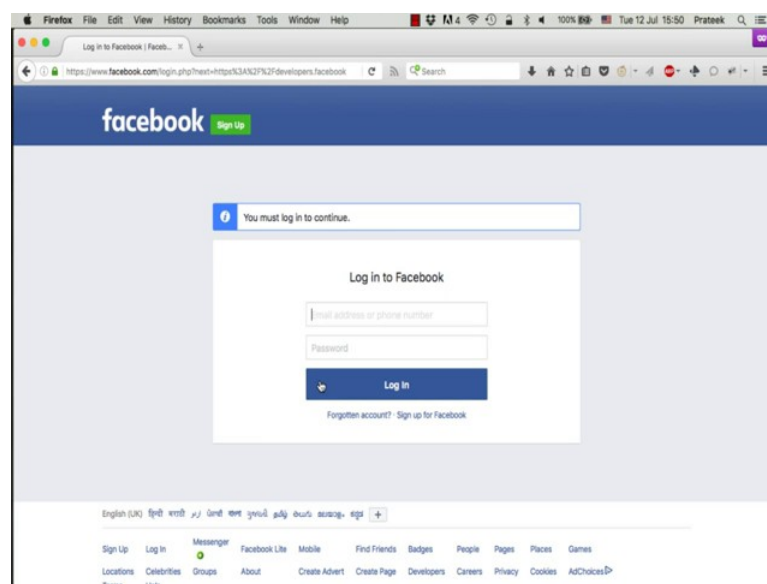


Now, click on tools and support and go the graph API explorer.

(Refer Slide Time: 01:06)

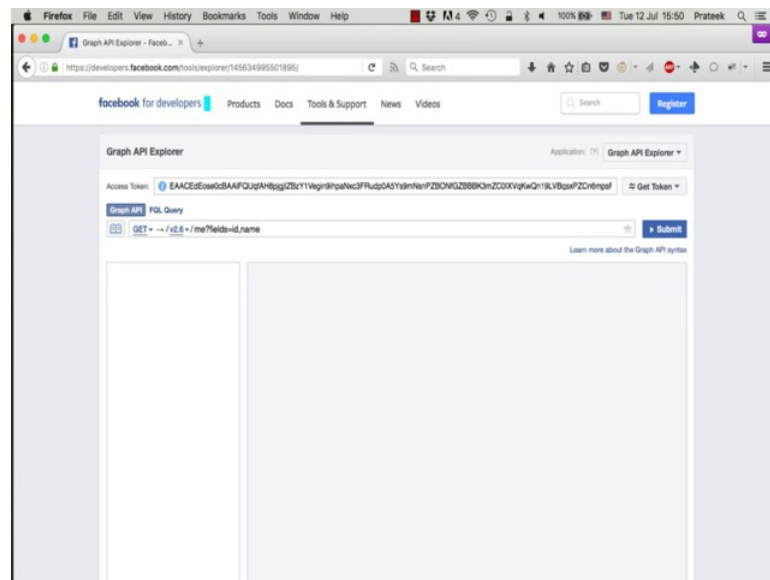


(Refer Slide Time: 01:12)



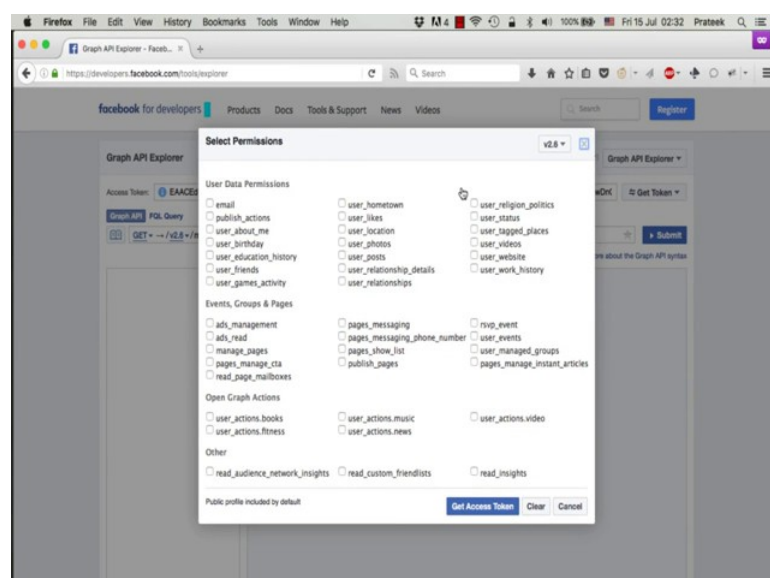
Of course, you need to login first. We assume all of you have an account on Facebook, if you do not please pause this tutorial here, create an account and then continue.

(Refer Slide Time: 00:12)

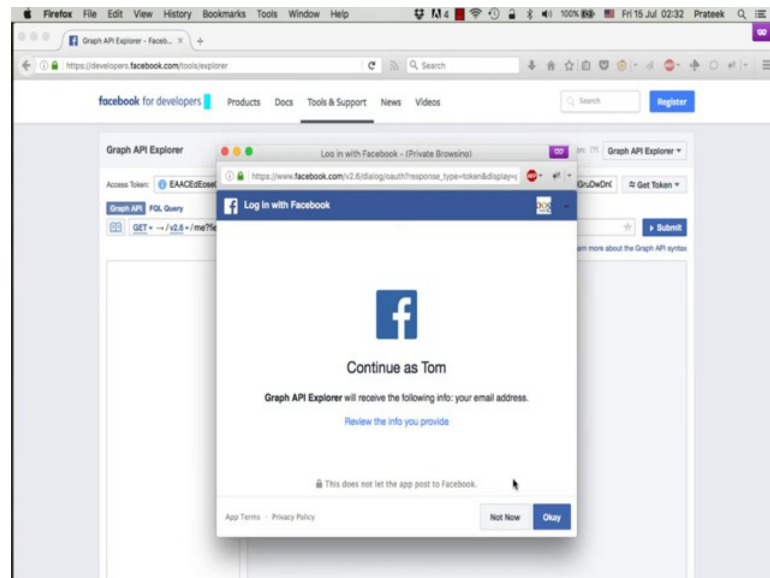


So, this is how the Facebook graph API explorer looks like. If you notice this bar here, you will see a long random looking string of letters and numbers; this is called the access token. Now an access token is like a key which opens the door of the Facebook API for you. In more technical terms, **the** access token is an authentication string generated using the open authentication or **OAuth** protocol which Facebook uses to verify the authenticity of the user and determine the user's identity. If you do not see an access token in this **space** click on the get access token button on the right and click on the get user access token option.

(Refer Slide Time: 02:05)

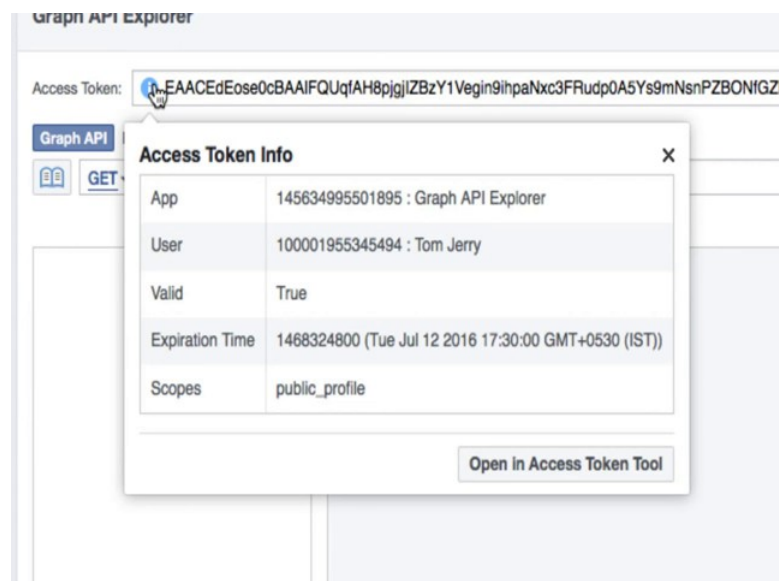


And you will see a big list of check boxes. Now each of these check boxes corresponds to a permission that you may choose to grant to Facebook. So, for now just click on any one of them say email and click on the get access token button. Do not worry about these permissions etcetera for now; we will get back to the details regarding all these permissions later in the tutorial.



A dialogue box will appear informing you that this action will allow Facebook to get to know your email address. Click on ok and you will have the access token appear in the access token bar.

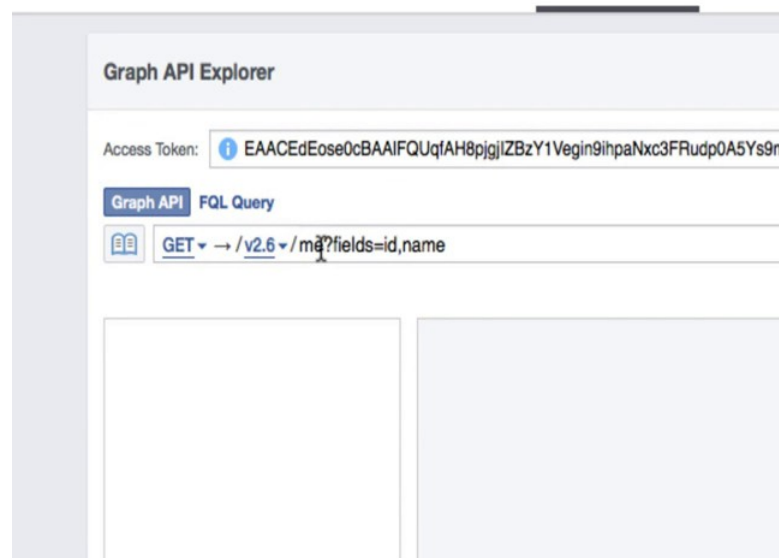
(Refer Slide Time: 02:48)



Now if you click on this little blue circle here, you can see more information about this access token. You can see the app that was used to generate this token it says the graph API explorer app here. Now this is the default app that is there when you first get into Facebook as a developer. Now this concept of apps is crucial from the developers' perspective. You can think of an app as a door that you need in order to enter the graph API. Now the key to this door is the access token like we just said before. So, essentially any access token or key corresponds to an app or a door which leads into the graph API. You cannot enter the graph API without a key that is the access token. And you cannot generate a key or an access token without a door that is the app. We will see how to create our own apps later in this tutorial.

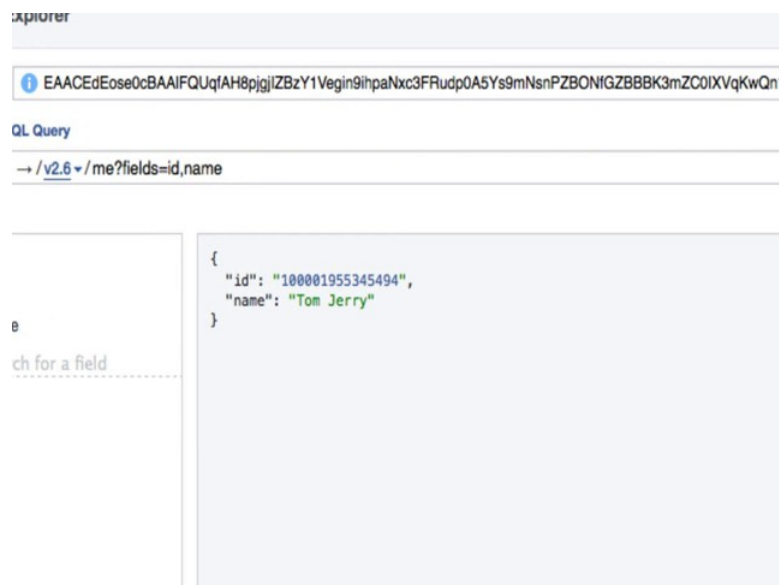
Now, getting back to the token, we can also see the name of the user who generated this token, which is Tom in this case. You see that the token is valid right now and the expiration time is about 17:30 hours which is less than 2 hours from now. So, after 2 hours, this access token becomes totally useless. Now all of this information is somehow embedded in this long random looking string. Now let us put this access token to use and extract some data from the API.

(Refer Slide Time: 04:04)



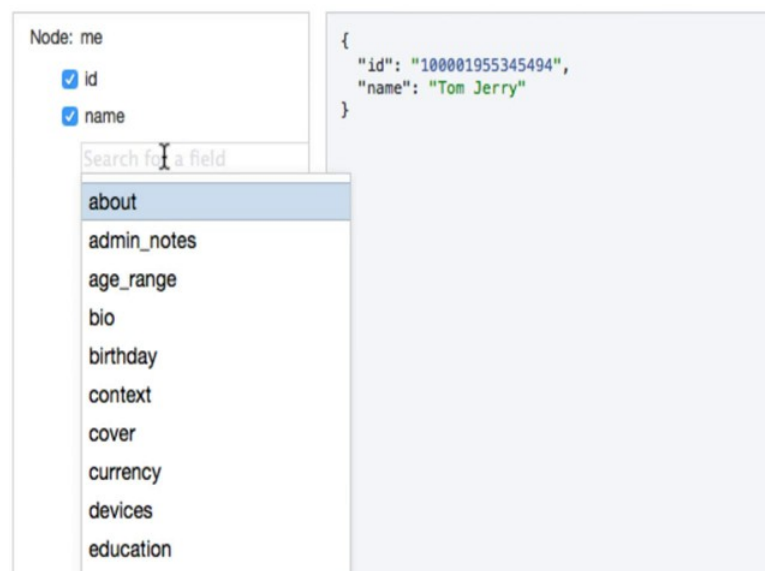
We see here that the query field has this me question mark fields is equal to id comma name here. So, just press, submit.

(Refer Slide Time: 04:12)



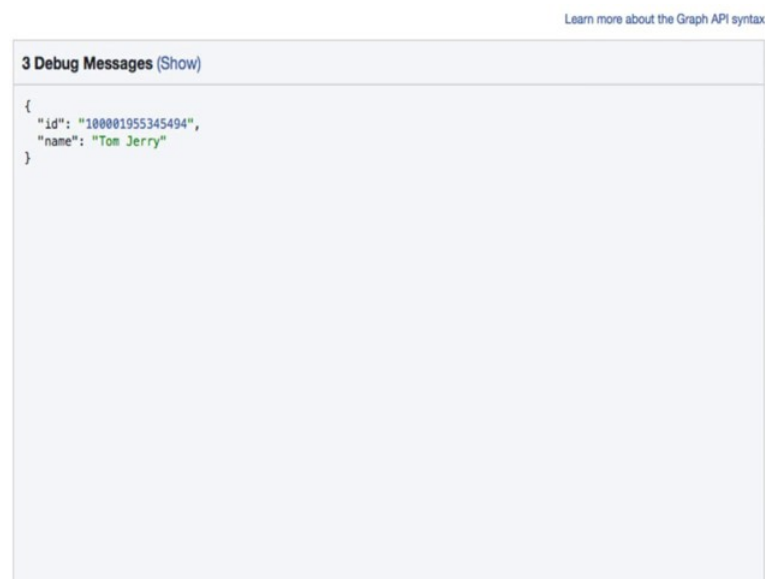
And you see that the API responds with the two fields that you asked for. The me part that we had in the query tells the API that it needs to return the id and name fields for me which is the current authenticating user. Now there are loads of other things that you can extract from the API.

(Refer Slide Time: 04:29)



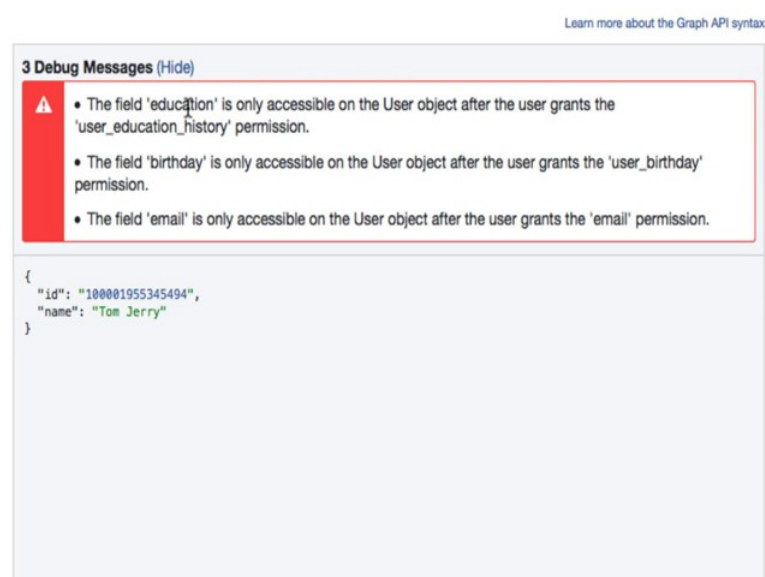
Just click on this search area and you will see a list to start with. So, let us select education, birthday, and say email and click submit again.

(Refer Slide Time: 04:39)



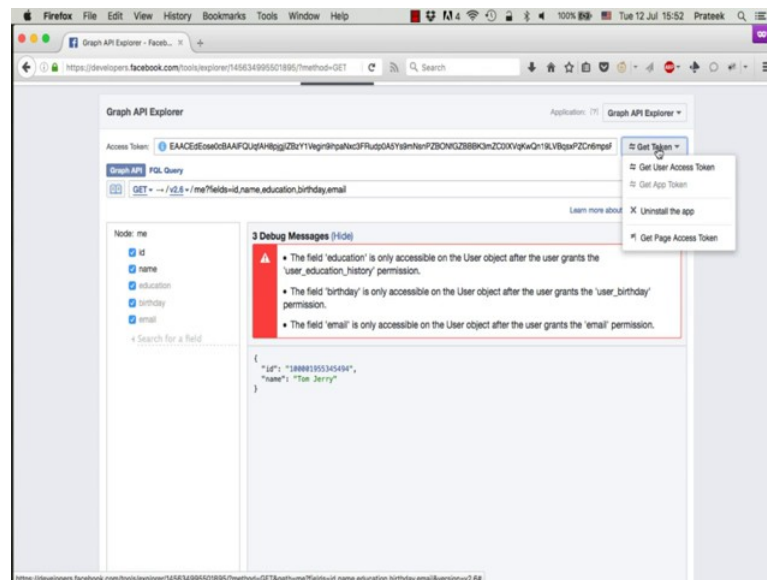
So, it throws a bunch of debug errors.

(Refer Slide Time: 04:44)



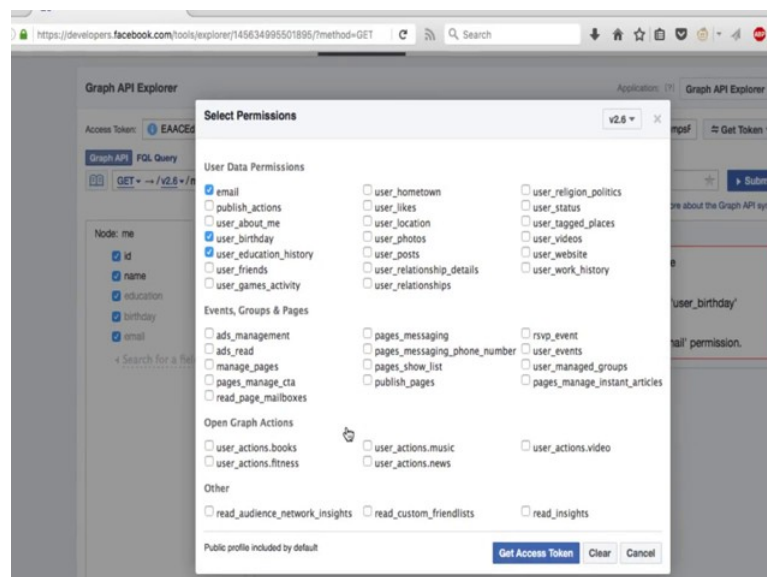
So, if you remember the permissions **a** while ago, we did not give permission to this app for getting access to birthday and its education fields.

(Refer Slide Time: 04:52)



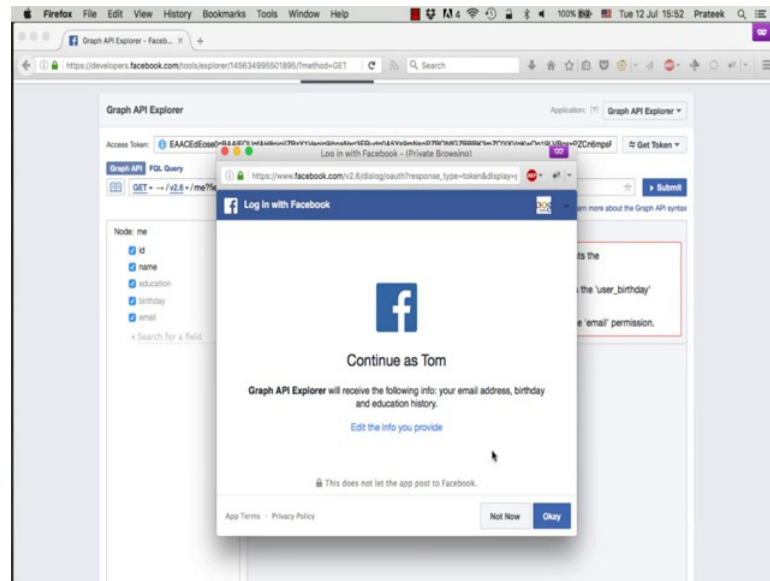
So, to get these fields from the API, go back to the get access token, click on the get user access token button again.

(Refer Slide Time: 04:55)



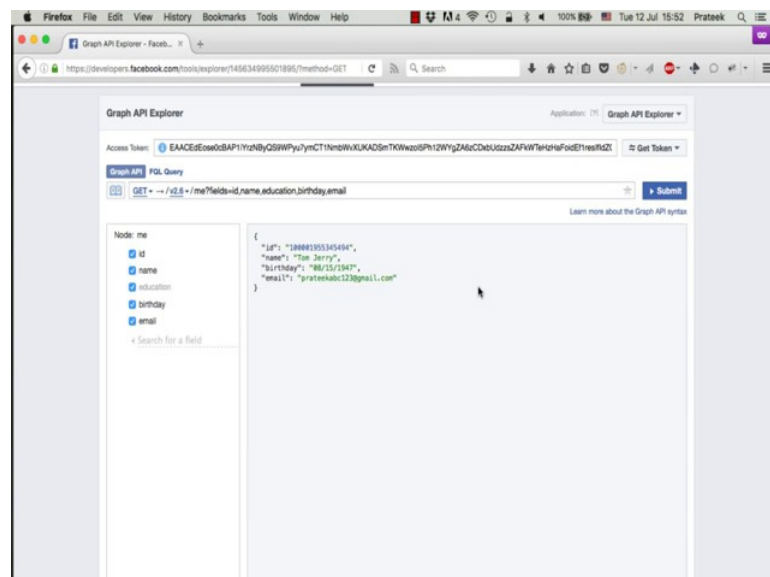
And this time check the 3 fields that you requested for, email, user education history and user birthday and click **get** access token.

(Refer Slide Time: 05:14)



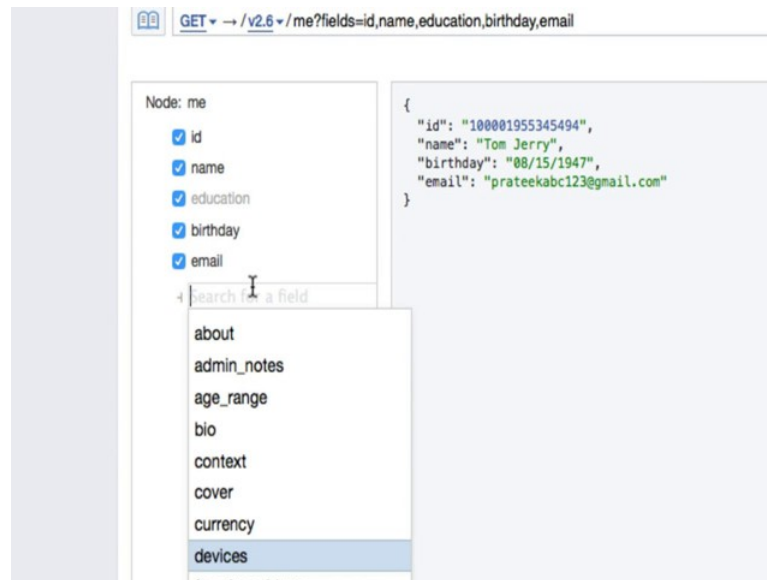
Now, you see this dialogue box asking if you want this app to have access to all these fields that you just checked. If you have ever used any app on Facebook before **FarmVille** or candy crush or any other app, you would remember seeing this exact dialogue box appear with probably a different set of permissions. This is the standard way all Facebook apps request permissions from a user. So, click OK.

(Refer Slide Time: 05:42)



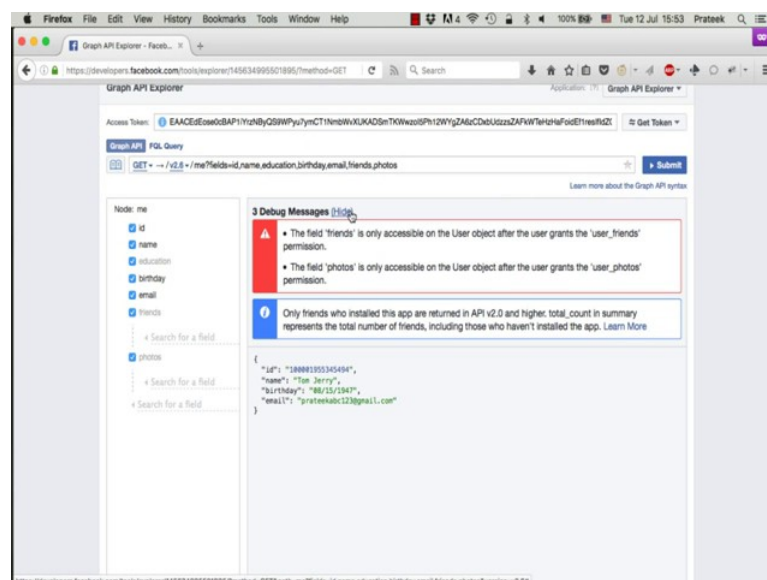
Click submit again and you will see that those debug messages are gone and you have the information you asked for. There is no education here because this user does not have any education history available in the profile.

(Refer Slide Time: 04:55)



So, let us look for some more details about this user say friends, photos and click submit.

(Refer Slide Time: 06:05)



Again, the same debug messages for the permissions.

(Refer Slide Time: 06:13)

User Data Permissions

<input checked="" type="checkbox"/> email x	<input type="checkbox"/> user_hometown	<input type="checkbox"/> user_religi
<input type="checkbox"/> publish_actions	<input type="checkbox"/> user_likes	<input type="checkbox"/> user_statu
<input type="checkbox"/> user_about_me	<input type="checkbox"/> user_location	<input type="checkbox"/> user_tagge
<input checked="" type="checkbox"/> user_birthday x	<input checked="" type="checkbox"/> user_photos	<input type="checkbox"/> user_video
<input checked="" type="checkbox"/> user_education_history x	<input type="checkbox"/> user_posts	<input type="checkbox"/> user_webs
<input checked="" type="checkbox"/> user_friends	<input type="checkbox"/> user_relationship_details	<input type="checkbox"/> user_work
<input type="checkbox"/> user_games_activity	<input type="checkbox"/> user_relationships	

Events, Groups & Pages

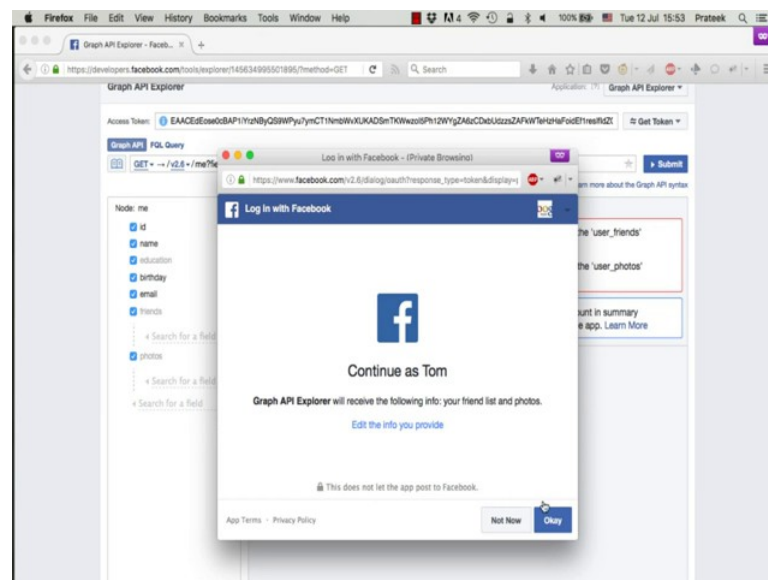
<input type="checkbox"/> ads_management	<input type="checkbox"/> pages_messaging	<input type="checkbox"/> rsvp_event
<input type="checkbox"/> ads_read	<input type="checkbox"/> pages_messaging_phone_number	<input type="checkbox"/> user_event
<input type="checkbox"/> manage_pages	<input type="checkbox"/> pages_show_list	<input type="checkbox"/> user_mana
<input type="checkbox"/> pages_manage_cta	<input type="checkbox"/> publish_pages	<input type="checkbox"/> pages_mar
<input type="checkbox"/> read_page_mailboxes		

Open Graph Actions

<input type="checkbox"/> user_actions.books	<input type="checkbox"/> user_actions.music	<input type="checkbox"/> user_actio
---------------------------------------------	---------------------------------------------	-------------------------------------

Go back to the get access token button select user photos and user friends, get access token, ok.

(Refer Slide Time: 06:24)



Submit.

(Refer Slide Time: 06:32)

```
{
  "id": "100001955345494",
  "name": "Tom Jerry",
  "birthday": "08/15/1947",
  "email": "prateekabc123@gmail.com",
  "friends": {
    "data": [
      {
        "name": "Prateek Dewan",
        "id": "517783245"
      }
    ]
  },
  "paging": {
    "cursors": {
      "before": "QVFjUjNRV3FpRwXON1pRNfdnc3pJU0k2Skk1c3ZA5X25yQXNydTvhb1pZA0DVKTVZAQZAS1IS05tSmtQnk1VeTlodG",
      "after": "QVFjUjNRV3FpRwXON1pRNfdnc3pJU0k2Skk1c3ZA5X25yQXNydTvhb1pZA0DVKTVZAQZAS1IS05tSmtQnk1VeTlodG"
    }
  },
  "summary": {
    "total_count": 6
  }
}
```

And there you go only one friend comes up, but notice that the total count says 6.

(Refer Slide Time: 06:39)

ame,education,birthday,email,friends,photos Submit

Learn more about the Graph API syntax

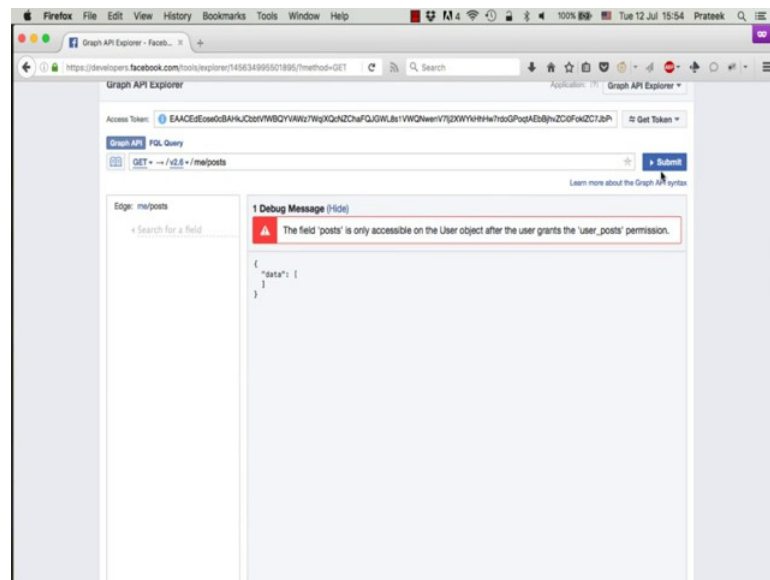
1 Debug Message (Hide)

Only friends who installed this app are returned in API v2.0 and higher. total_count in summary represents the total number of friends, including those who haven't installed the app. [Learn More](#)

```
{
  "id": "100001955345494",
  "name": "Tom Jerry",
  "birthday": "08/15/1947",
  "email": "prateekabc123@gmail.com",
  "friends": {
    "data": [
      {
        "name": "Prateek Dewan",
        "id": "517783245"
      }
    ]
  },
  "paging": {
    "cursors": {
      "before": "QVFjUjNRV3FpRwXON1pRNfdnc3pJU0k2Skk1c3ZA5X25yQXNydTvhb1pZA0DVKTVZAQZAS1IS05tSmtQnk1VeTlodG",
      "after": "QVFjUjNRV3FpRwXON1pRNfdnc3pJU0k2Skk1c3ZA5X25yQXNydTvhb1pZA0DVKTVZAQZAS1IS05tSmtQnk1VeTlodG"
    }
  },
  "summary": {
    "total_count": 6
  }
}
```

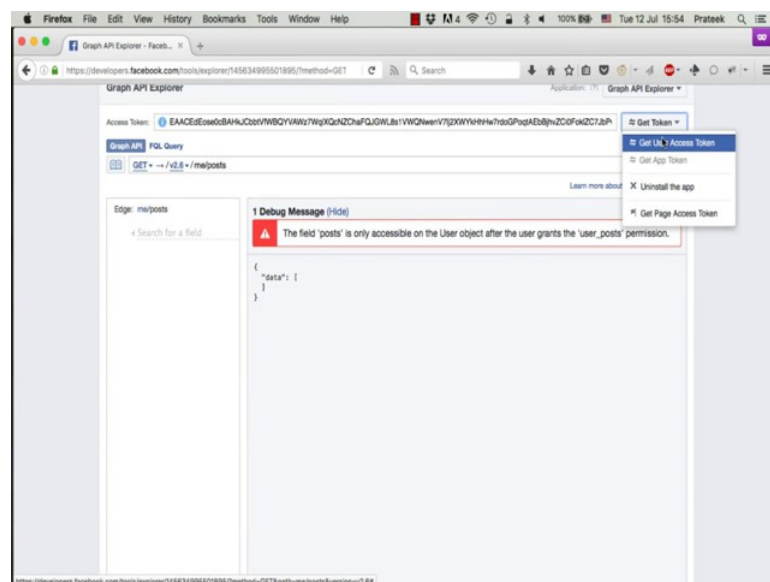
This debug message here gives the explanation for this difference. So, it says that only those friends are returned by the API who have installed or used this graph API explorer app. So, which means that this one friend that showed up in the result has used a graph API explorer, but the other 5 have not. Facebook did return the details of all the friends up until last year, but due to multiple privacy issues raised by some users, they stopped doing it.

(Refer Slide Time: 07:11)

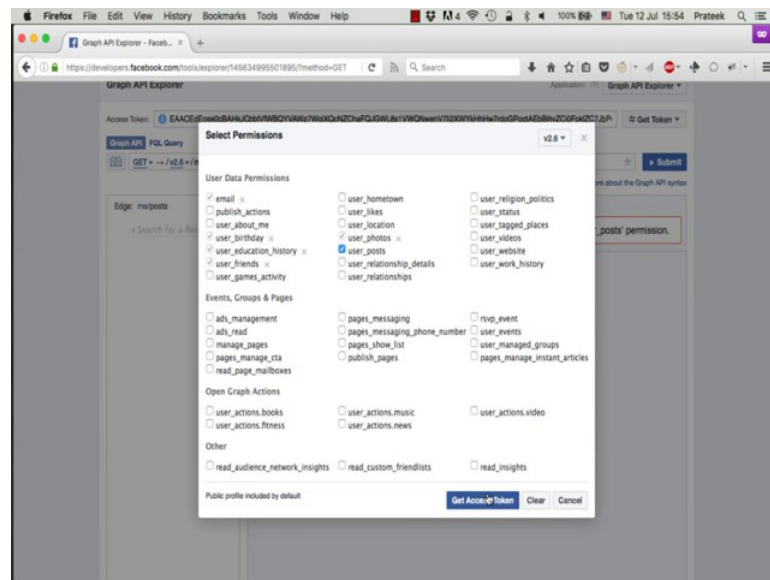


Anyway, so now, let us look at the posts that this user has done. Go to the query area remove the existing query and just type me slash posts and press enter. Again, the same rule.

(Refer Slide Time: 07:17)



(Refer Slide Time: 07:21)



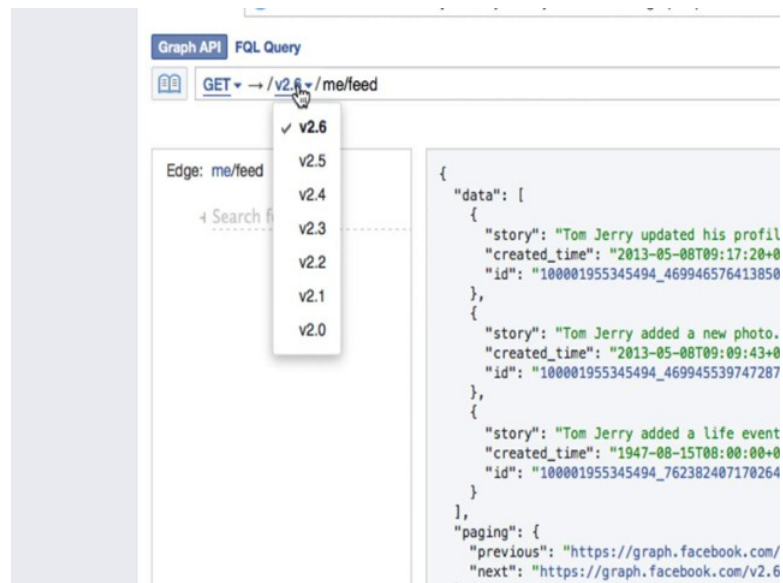
So, let us get the user underscore post permissions and there you go.

(Refer Slide Time: 07:30)



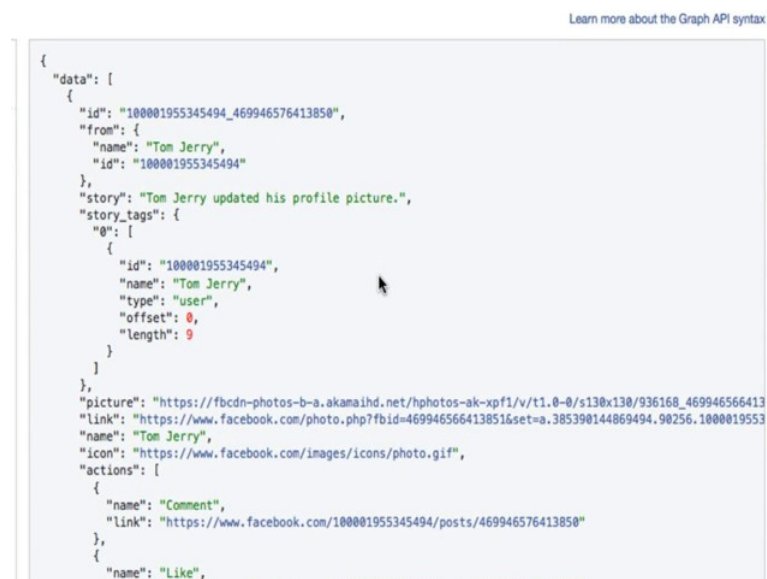
So, we see that the user updated his profile picture on May 8, 2013.

(Refer Slide Time: 07:36)



Now if you notice here this part says v 2.6. This is the latest version 2.6 of the API that we are currently using. So, Facebook regularly pushes out newer versions of the graph API every few months and gets rids of the old ones over time. For now, these are the versions available. Now let us check out what is the difference between this current version 2.6 and the oldest available version, which is 2.0, right.

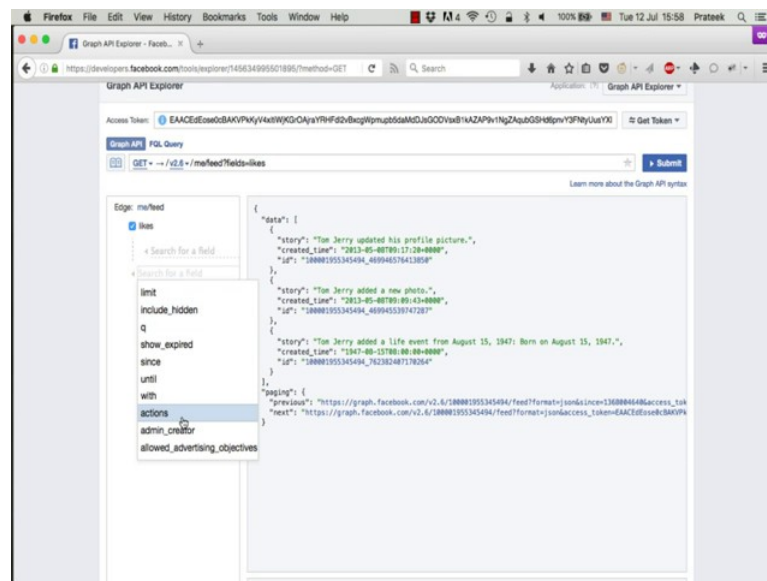
(Refer Slide Time: 08:05)



Now just select the 2.0 from the drop down and press submit. And you see that there is a whole lot of extra information that **shows** up. We **are** still looking at the posts, but now

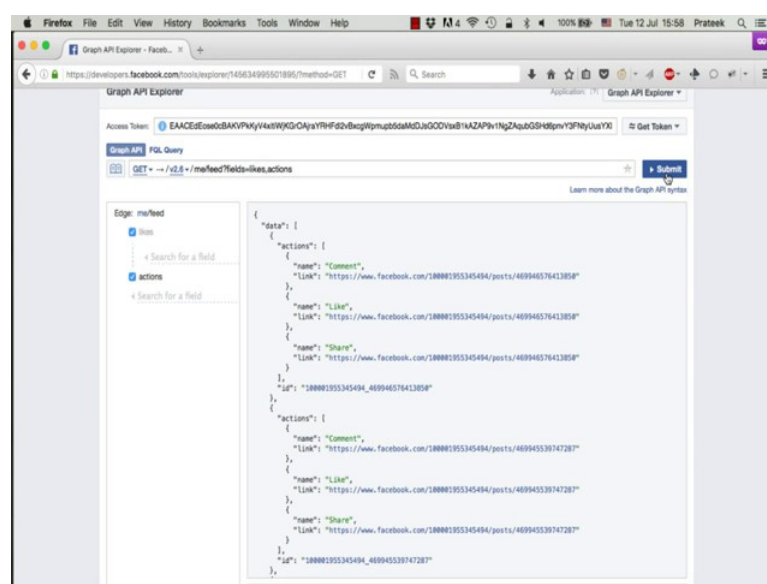
there is much more information about the post available. Like the link of the pictures the tags, the privacy settings for the post, the type of the post which is photo here, and multiple other things. It is the same level of details for all the posts. Now if you go back to the latest version 2.6, you notice that all these fields are still available, but you need to explicitly request for these fields they are not returned by the default request.

(Refer Slide Time: 08:39)



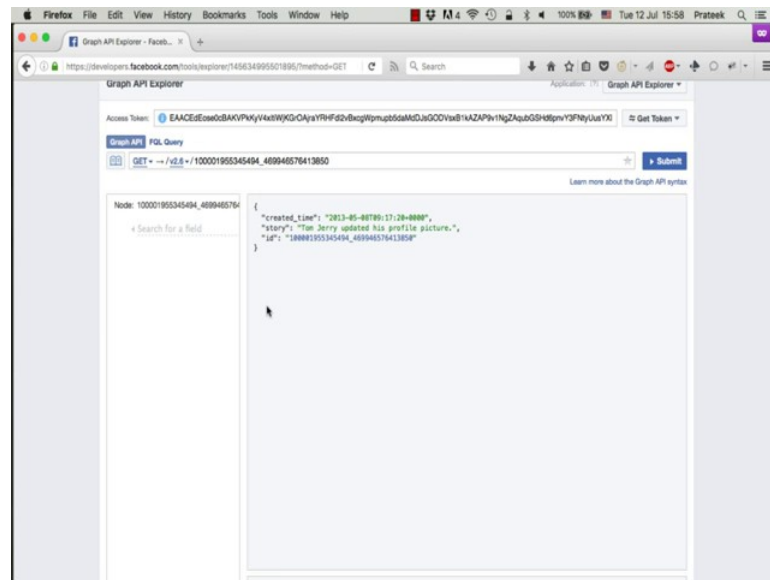
So, you select likes, let us say actions and click submit.

(Refer Slide Time: 08:45)



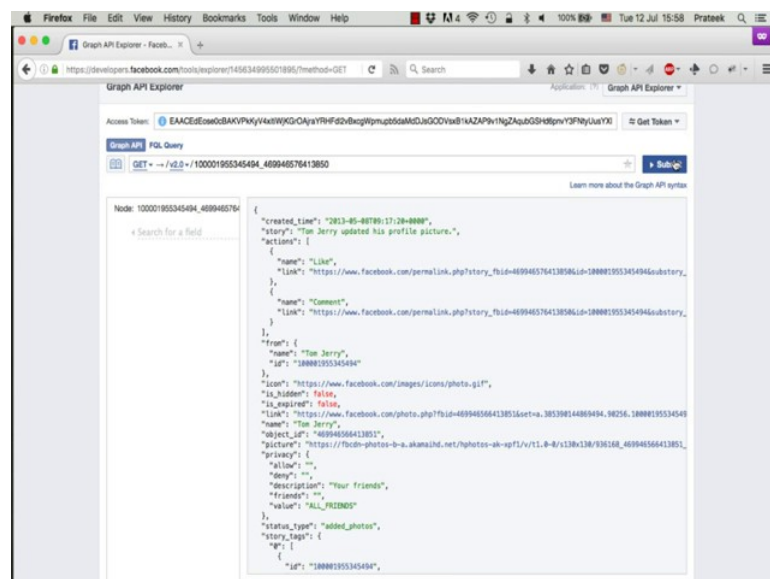
And you can get these fields. So, it looks like there are 0 likes on all the user's post. So, they are not showing up. You can also look at individual post using their id; this id is unique for every posts made by any user on Facebook.

(Refer Slide Time: 09:03)



So, you click on id, and you get just this post. Again you can get more details if you use the version 2.0.

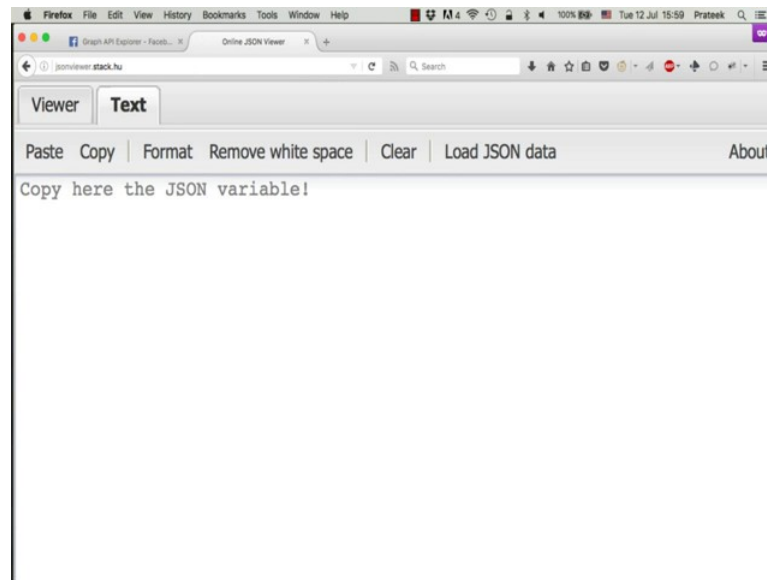
(Refer Slide Time: 09:14)



Now the graph API returns the data in a json format, which is javascript object notation format comprising of multiple curly brackets and square brackets. This json format can

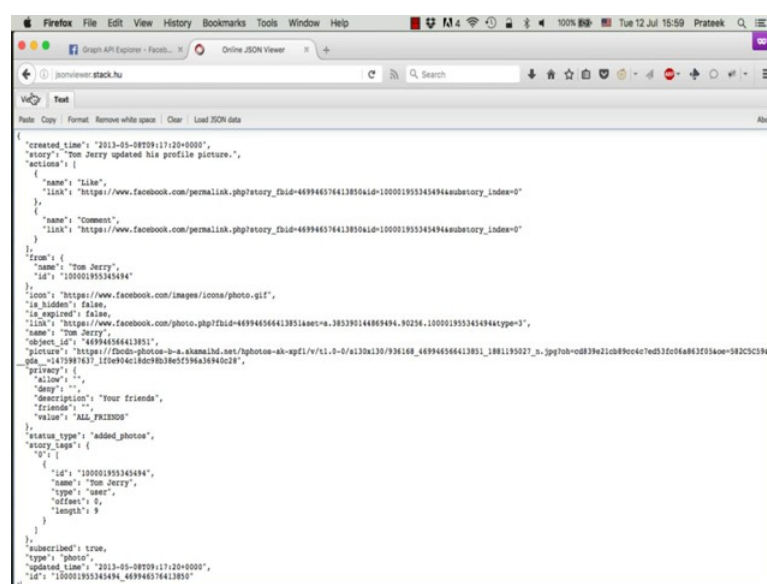
be hard to read sometimes. So, to make it easier to read and understand, just copy this entire **response**, and make sure you do not miss the starting or the ending brackets that just destroys the entire format of data. So, you select this, copy it.

(Refer Slide Time: 09:49)



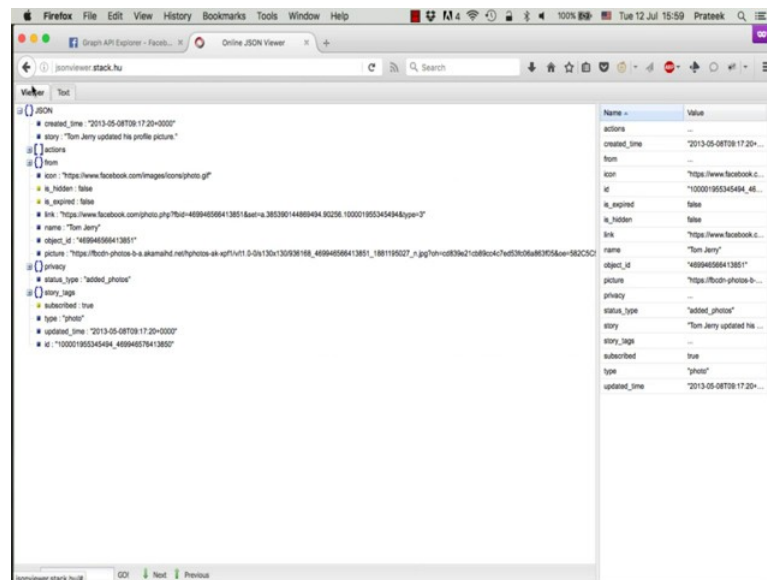
And we will now see this data in the more readable format using a tool called json viewer. There are many such tools available online for free. The one we will be using is **at** json viewer dot stack dot hu.

(Refer Slide Time: 09:51)



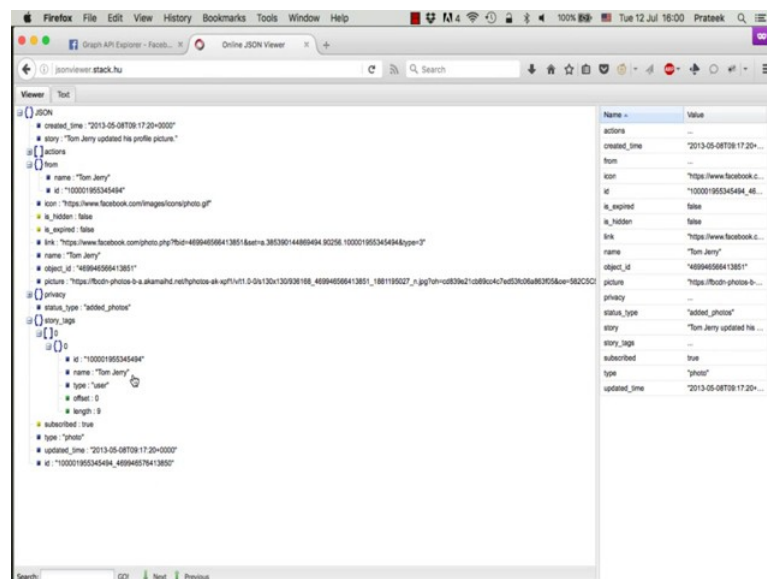
So, just paste this data that you copied.

(Refer Slide Time: 09:57)



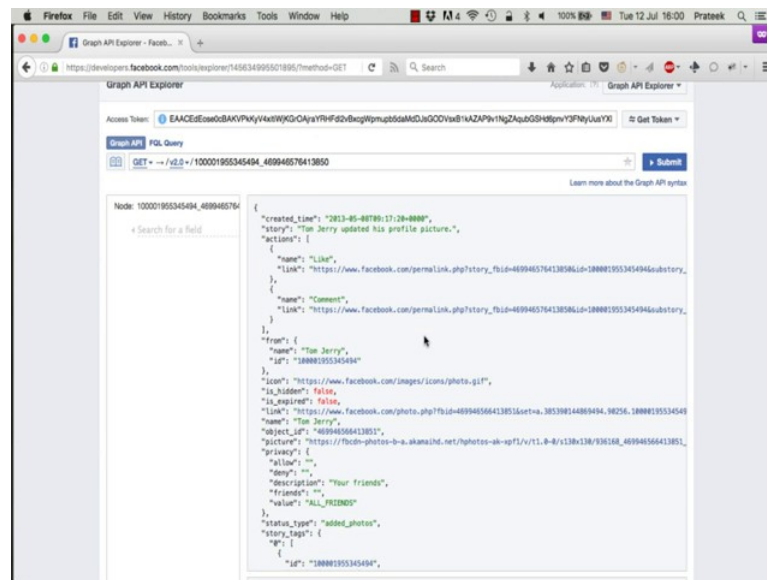
Go to the viewer tab and you can see the same data

(Refer Slide Time: 10:02)



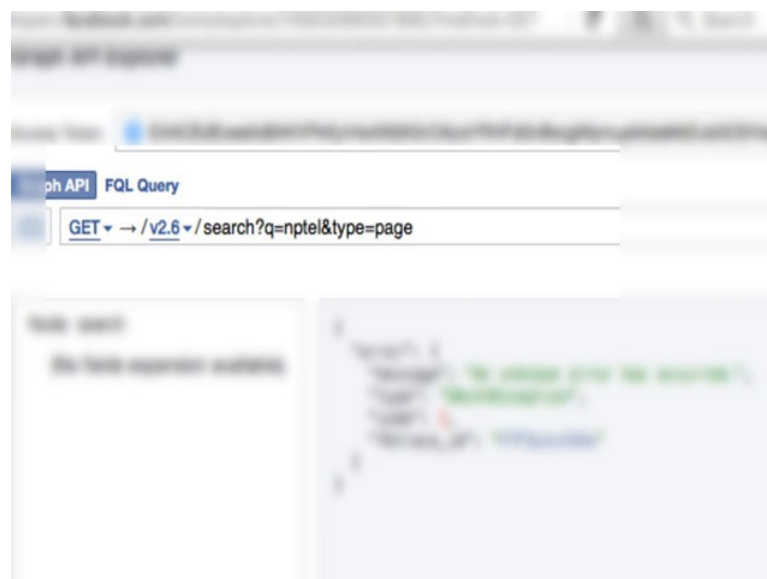
But it is much more easy to understand and easy to read. You can easily open and close brackets to view the data inside them. It just makes life a lot easier.

(Refer Slide Time: 10:20)



Now you can also perform search operations using the graph API to search for users, pages, groups, events etcetera on Facebook, but note that the search only **returns** results whose privacy settings are set to public.

(Refer Slide Time: 10:35)



So, the search query has a minimum of two parameters a query parameter q lets search for npTEL and a type parameter which tells the API what type of results to look for. So, let us say page. So, we are essentially searching for Facebook pages named npTEL.

(Refer Slide Time: 10:53)

```
{
  "data": [
    {
      "name": "NPTEL",
      "id": "135684199798143"
    },
    {
      "name": "NPTEL",
      "id": "1413735098927291"
    },
    {
      "name": "NPTEL HRD Computer Science videos",
      "id": "1035573326474138"
    },
    {
      "name": "NPTEL",
      "id": "1607438936163989"
    },
    {
      "name": "NPTEL - A Joint Initiative of IITs and IISc",
      "id": "534937499965999"
    },
    {
      "name": "NPTEL Mechanical",
      "id": "435380716603755"
    },
    {
      "name": "NPTEL Local Chapter - B8IT",
      "id": "1717301228550688"
    },
    {
      "name": "NPTEL Condenser",
      "id": "103393996711333"
    },
    {
      "name": "NPTEL IISc",
      "id": "103393996711333"
    }
  ]
}
```

So, see we got a whole list of pages with npTEL in their name.

(Refer Slide Time: 11:03)

developers.facebook.com/tools/explorer/145634995501895/?method=GET

Graph API Explorer

Access Token: EAACEdEose0cBAKVpKkyV4xitiWjKGroAjaYRHFdi2vBxcgWpmupb5daMdDJsGODVsx81kAZAP9v1NgZA

Graph API FQL Query

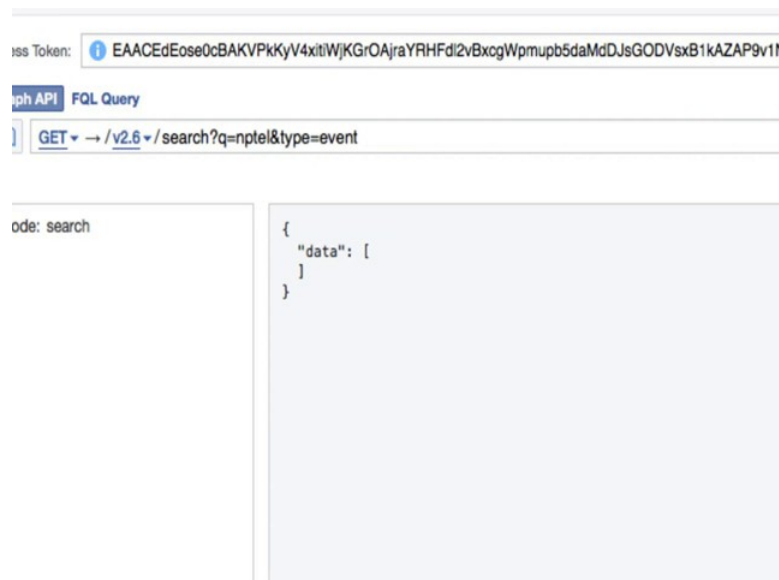
GET → /v2.6/search?q=npTEL&type=group

Node: search

```
{
  "data": [
    {
      "name": "NPTEL",
      "id": "135684199798143"
    },
    {
      "name": "NPTEL",
      "id": "1413735098927291"
    },
    {
      "name": "NPTEL HRD Computer Science videos",
      "id": "1035573326474138"
    },
    {
      "name": "NPTEL",
      "id": "1607438936163989"
    },
    {
      "name": "NPTEL - A Joint Initiative of IITs and IISc",
      "id": "534937499965999"
    },
    {
      "name": "NPTEL Mechanical",
      "id": "435380716603755"
    },
    {
      "name": "NPTEL Local Chapter - B8IT",
      "id": "1717301228550688"
    },
    {
      "name": "NPTEL Condenser",
      "id": "103393996711333"
    },
    {
      "name": "NPTEL IISc",
      "id": "103393996711333"
    }
  ]
}
```

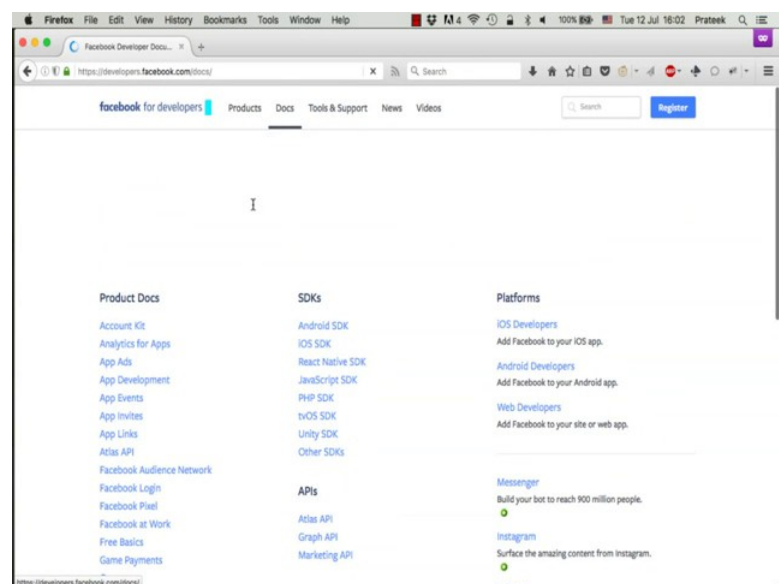
Now, let us look for groups named npTEL. So, you change the type to be group press enter and there is a bunch of groups with npTEL in their name. You can see **that** some of these groups are closed and some of them are open.

(Refer Slide Time: 11:17)



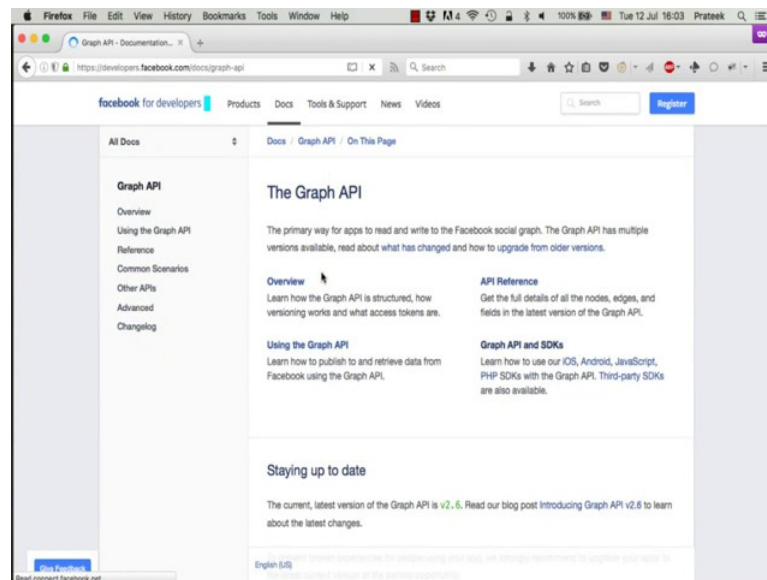
Let us look for events now. So, type is equal to event. So, there are no Facebook events in the name of npitel right now.

(Refer Slide Time: 11:30)



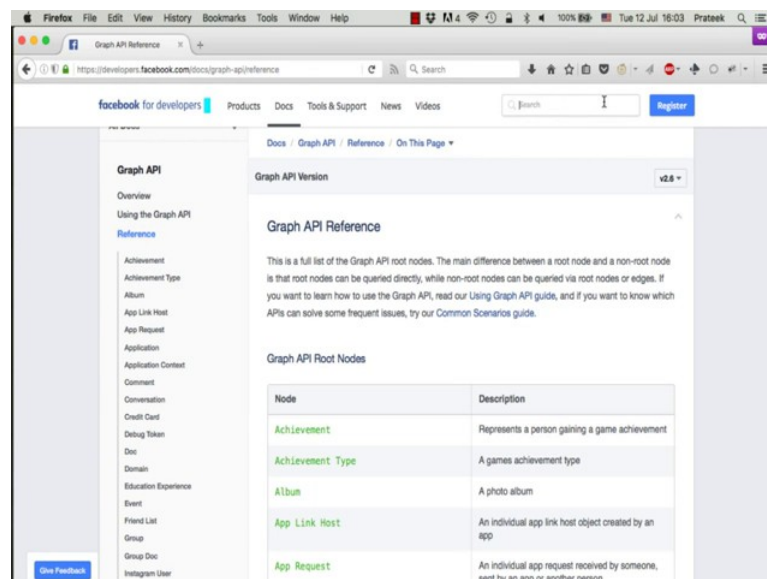
So, Facebook provides this whole set of documentation that you can refer to, to learn what all you can do with the graph API.

(Refer Slide Time: 11:37)



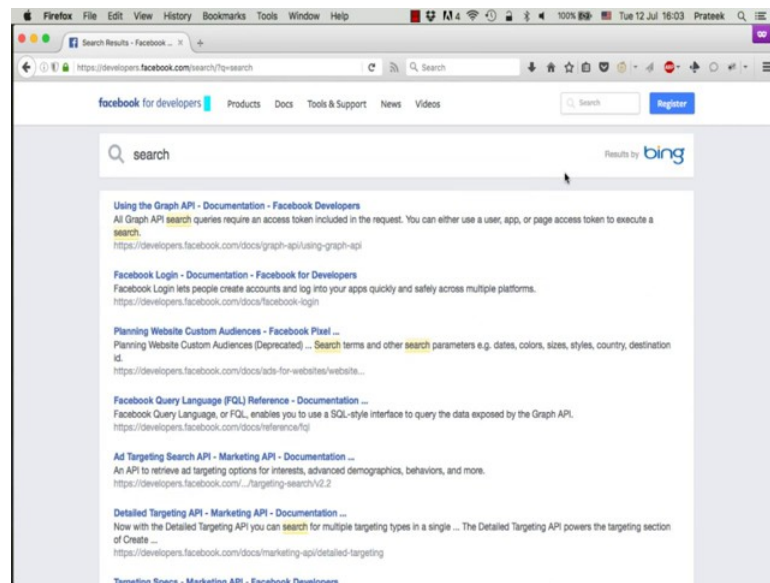
Just go to docs and click on graph API.

(Refer Slide Time: 11:50)



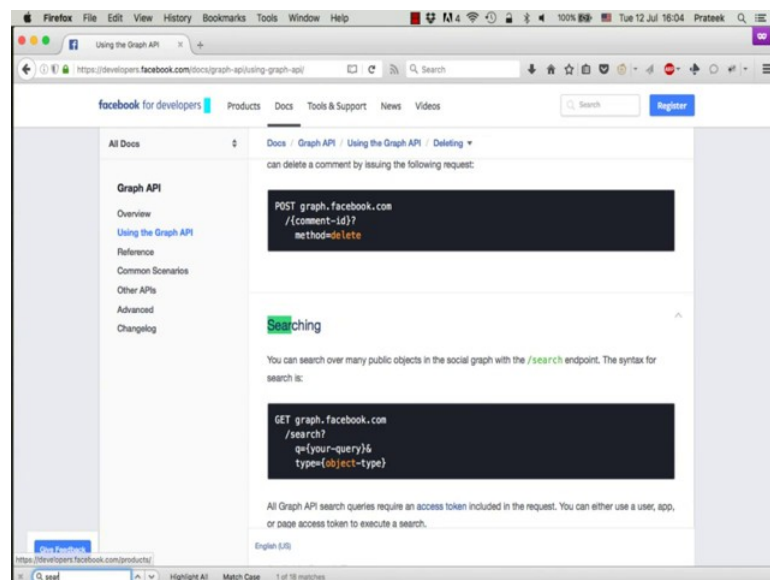
You can also search the documentation for what you need.

(Refer Slide Time: 11:56)



For example, let us look for help on the search query that **we** just used to search for nptel pages.

(Refer Slide Time: 12:01)



So, this section shows you how to use graph API search.

(Refer Slide Time: 12:11)

Searching

You can search over many public objects in the social graph with the `/search` endpoint. The syntax for search is:

```
GET graph.facebook.com
/search?
q={your-query}&
type={object-type}
```

All Graph API search queries require an [access token](#) included in the request. You can either use a user, app, or page access token to execute a search.

Available Search Types

We support search for the following types:

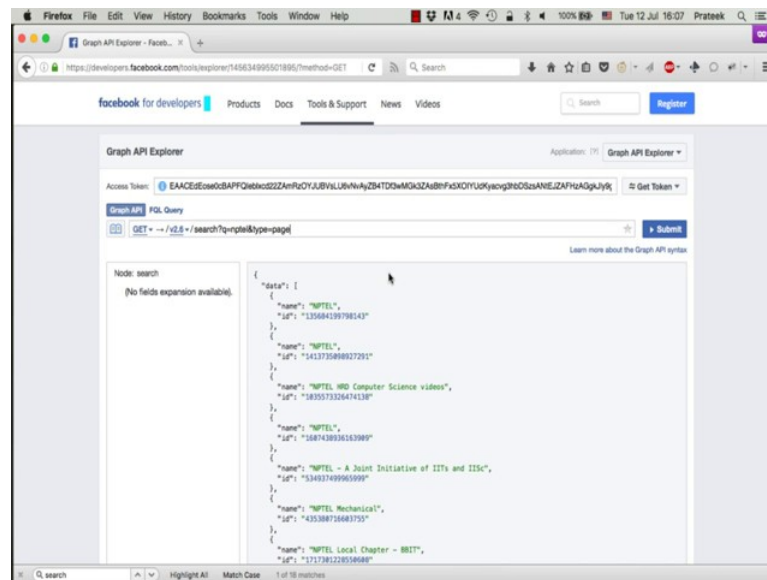
This is exactly how we did it. Search question mark `q` is equal to query and `Type` is equal to object type.

(Refer Slide Time: 12:17)

<code>user</code>	Search for a person (if they allow their name to be searched for).	Name
<code>page</code>	Search for a Page.	Name
<code>event</code>	Search for an event.	Name
<code>group</code>	Search for a Group.	Name
<code>place</code>	Search for a place. You can narrow your search to a specific location and distance by adding the <code>center</code> parameter (with latitude and longitude) and an optional <code>distance</code> parameter (in meters):	Name

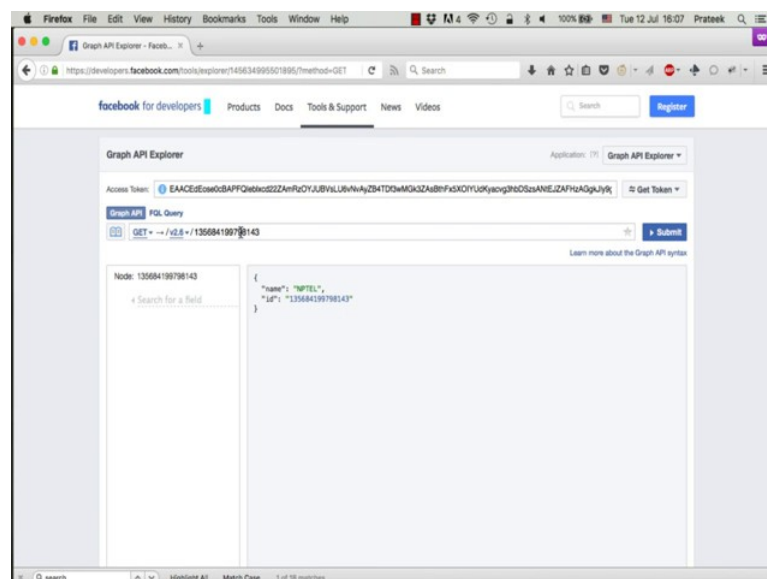
If you scroll down a bit, you can see the various type of objects you can search for. Users, pages, events, groups, places etcetera.

(Refer Slide Time: 12:28)



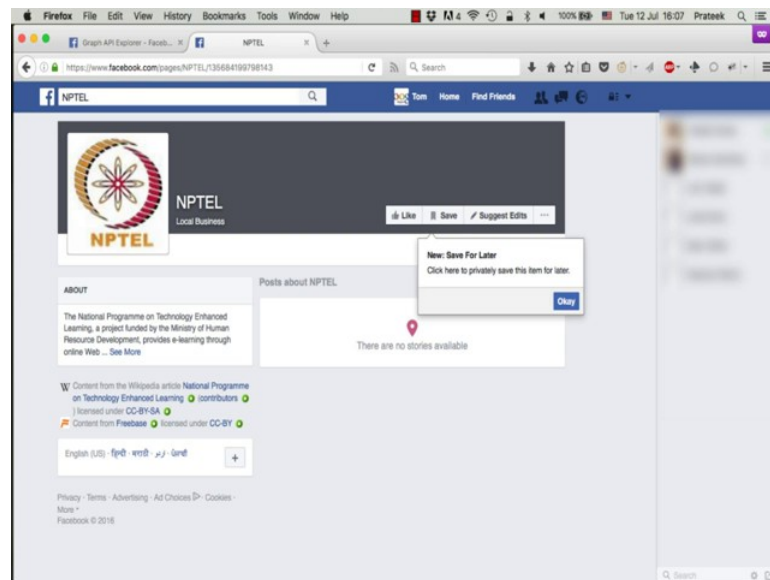
So, now let us get back to the graph API. Now when we search for pages with the name nptel, it is impossible to tell anything about the pages which show up by just looking at the search results. So, let us look at these pages manually one by one.

(Refer Slide Time: 12:45)



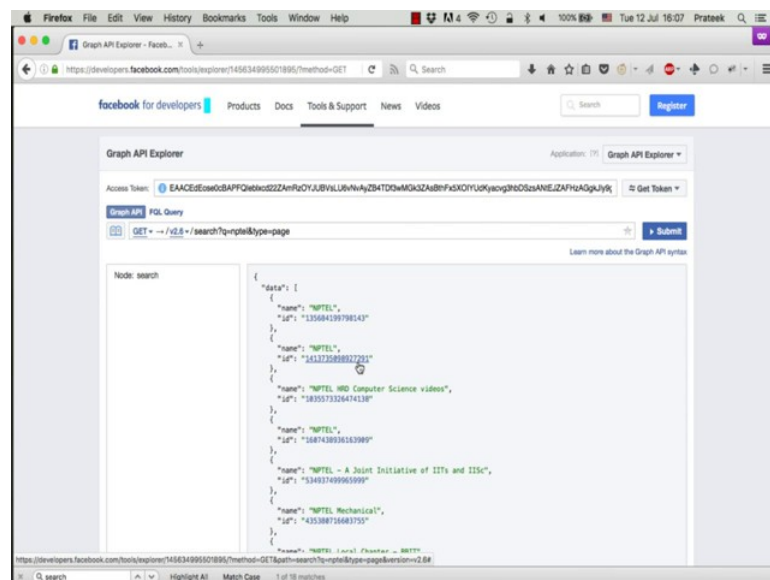
Click on the id field of the first search result. Copy this id and open Facebook dot com slash this id in a new tab this will open up the actual Facebook page corresponding to this id.

(Refer Slide Time: 12:56)



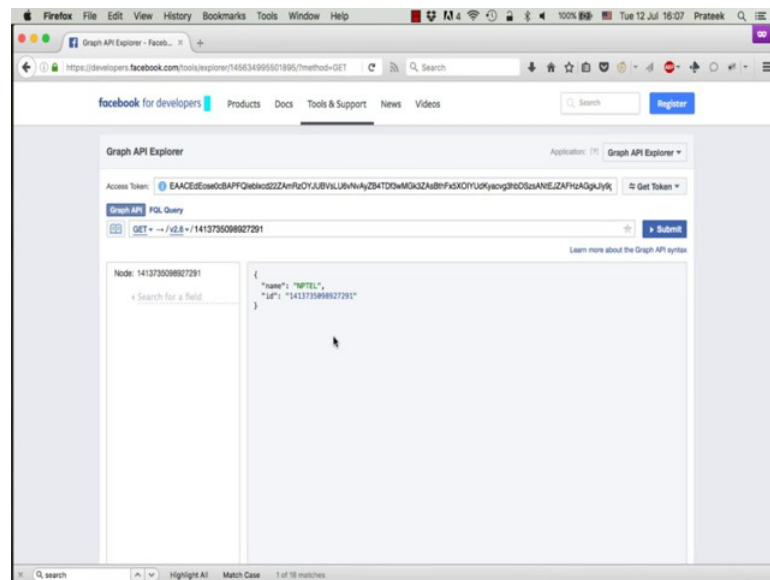
Now if you notice this is an empty page with no posts, no real information no likes. So, this is definitely not the real npTEL page.

(Refer Slide Time: 13:07)



So, now, let us look at the next result; same process.

(Refer Slide Time: 13:18)



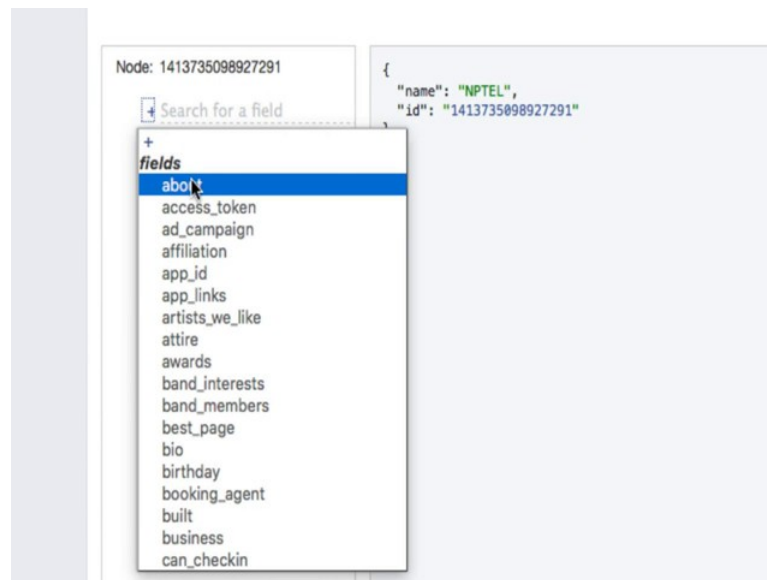
Click on the id of the next result copy this id and open Facebook dot com slash id in a new tab.

(Refer Slide Time: 13:25)



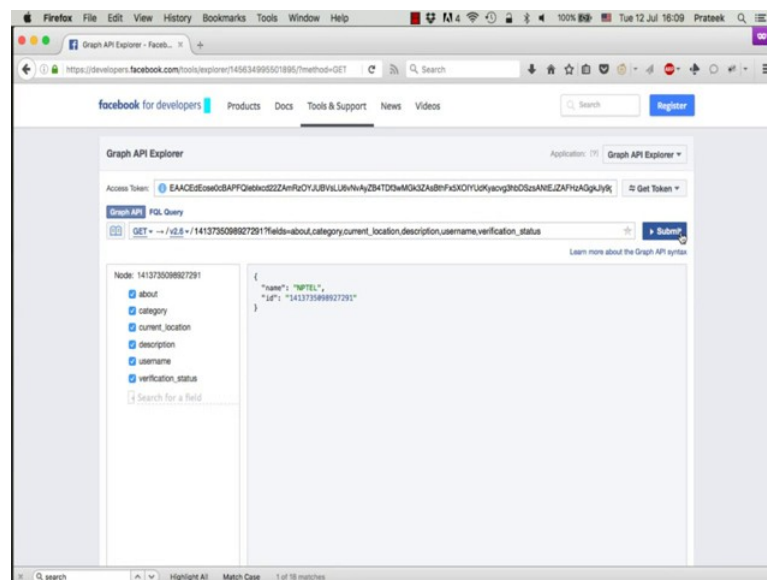
So, this looks much more like a genuine npTEL page. It is rated 4.9 on 5, close to 10000 likes; the descriptions says it is very responsive to messages. So, now let us try to extract data from this page using the API.

(Refer Slide Time: 13:53)



Let us try to get the about field and say category of the page, then current location, description, username and verification status.

(Refer Slide Time: 14:12)

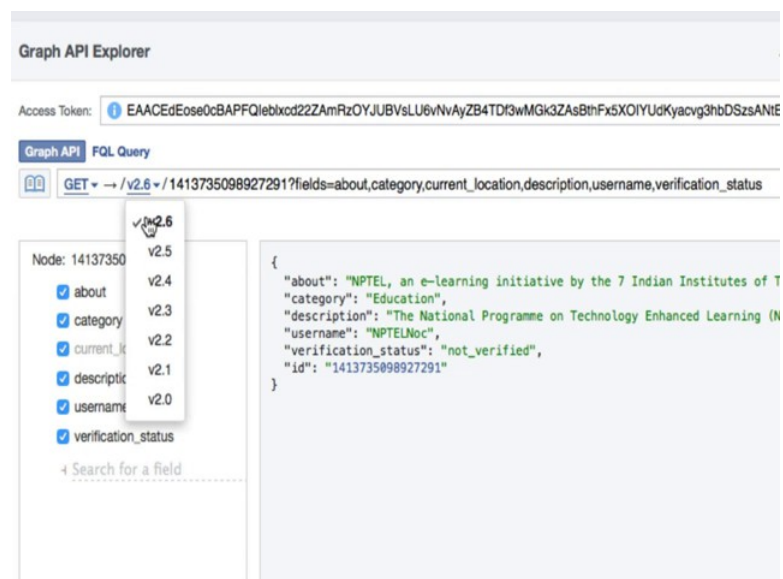


(Refer Slide Time: 14:15)



And there you see the page is not verified and all the other bunch of information we requested for. Now, let us switch to version 2.0 of the API, like we did a few minutes ago and see what all information is available for the page.

(Refer Slide Time: 14:28)



Now select 2.0 from the drop down remove these fields and click submit.


(Refer Slide Time: 14:36)



```
{
  "id": "1413735098927291",
  "about": "NPTEL, an e-learning initiative by the 7 Indian Institutes of Technology and the Indian Institute of Technology",
  "can_post": false,
  "category": "Education",
  "category_list": [
    {
      "id": "151676848228295",
      "name": "Education"
    }
  ],
  "checkins": 37,
  "cover": {
    "cover_id": "1414194822214732",
    "offset_x": 0,
    "offset_y": 75,
    "source": "https://fbcdn-sphotos-e-a.akamaihd.net/hphotos-ak-xaf1/v/t1.0-9/10361315_1414194822214732_58211414194822214732",
    "id": "1414194822214732"
  },
  "description": "The National Programme on Technology Enhanced Learning (NPTEL) is a collaborated initiative",
  "founded": "7 IITs and IISc, supported by MHRD, Govt of India from 2003",
  "has_added_app": false,
  "hours": {
    "mon_1_open": "09:00",
    "mon_1_close": "18:00",
    "tue_1_open": "09:00",
    "tue_1_close": "18:00",
    "wed_1_open": "09:00",
    "wed_1_close": "18:00",
    "thu_1_open": "09:00",
    "thu_1_close": "18:00",
    "fri_1_open": "09:00",
    "fri_1_close": "18:00",
    "sat_1_open": "09:00",
    "sat_1_close": "18:00"
  },
  "is_community_page": false,
  "is_published": true,
  "likes": 9718,
  "link": "https://www.facebook.com/NPTELNoc/",
  "location": null
}
```

And look at this, you have category, check-ins, details about the cover picture of the page, opening and closing hours, location with latitude and longitude, phone number, number of people who liked this page and so on.

(Refer Slide Time: 15:01)



```
GET /v2.0/1413735098927291/

le: 1413735098927291

Search for a field

founded : 7 IITs and IISc, supported by MHRD, Govt of India
"has_added_app": false,
"hours": {
  "mon_1_open": "09:00",
  "mon_1_close": "18:00",
  "tue_1_open": "09:00",
  "tue_1_close": "18:00",
  "wed_1_open": "09:00",
  "wed_1_close": "18:00",
  "thu_1_open": "09:00",
  "thu_1_close": "18:00",
  "fri_1_open": "09:00",
  "fri_1_close": "18:00",
  "sat_1_open": "09:00",
  "sat_1_close": "18:00"
},
"is_community_page": false,
"is_published": true,
"likes": 9718,
"link": "https://www.facebook.com/NPTELNoc/",
"location": null
}
```

Now you can also get the page feed meaning all the posts that appear on the page when you open it. In the query bar, just add slash feed after the page id and press enter.

(Refer Slide Time: 15:06)

```
/feed
{
  "data": {
    {
      "id": "1413735098927291_1590732211227578",
      "from": {
        "name": "NPTEL",
        "category": "Education",
        "category_list": [
          {
            "id": "151676048228295",
            "name": "Education"
          }
        ]
      },
      "id": "1413735098927291"
    },
    {
      "message": "Android Skilling and Certification Programme launched by #GoogleIndia and available for free",
      "picture": "https://fbcdn-photos-b-a.akamaihd.net/hphotos-ak-xfp1/v/t1.0-0/s130x130/13627126_1590732211",
      "link": "https://www.facebook.com/NPTELNOC/photos/a.1422957880805626.1873741828.1413735098927291/159073",
      "name": "Timeline Photos",
      "icon": "https://www.facebook.com/images/icons/photo.gif",
      "actions": [
        {
          "name": "Comment",
          "link": "https://www.facebook.com/1413735098927291/posts/1590732211227578"
        },
        {
          "name": "Like",
          "link": "https://www.facebook.com/1413735098927291/posts/1590732211227578"
        }
      ]
    },
    {
      "privacy": {
        "value": "",
        "description": "",
        "friends": "",
        "allow": "",
        "deny": ""
      }
    }
  ]
}
```

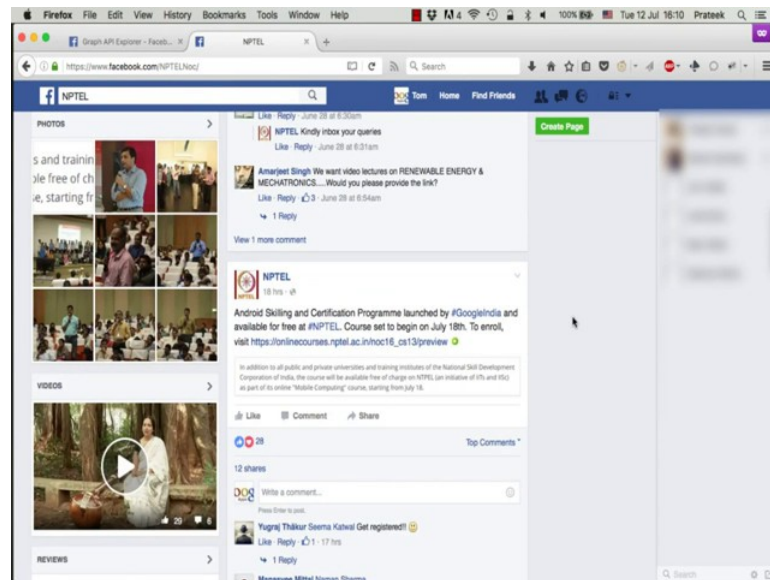
So, you see a list of all the posts that the page has done and all the posts that any other Facebook user or page has done on this npTEL page here. This usually appears in reverse chronological order; meaning most recent posts come first. The first most recent post says something about android skills and certification programme launched by Google India.

(Refer Slide Time: 15:35)



You can verify it is the same content as on the page. Let us go back to the page. So, this first post is actually a pinned post. Page administrators can pin a particular post, which they want everyone to see whenever they visit the page at the top.

(Refer Slide Time: 15:48)



But if you scroll down a bit you will see the remaining posts in the most recent first order. So, here it is the same post that we saw in the API, the android skill and certification programme post.

(Refer Slide Time: 16:03)



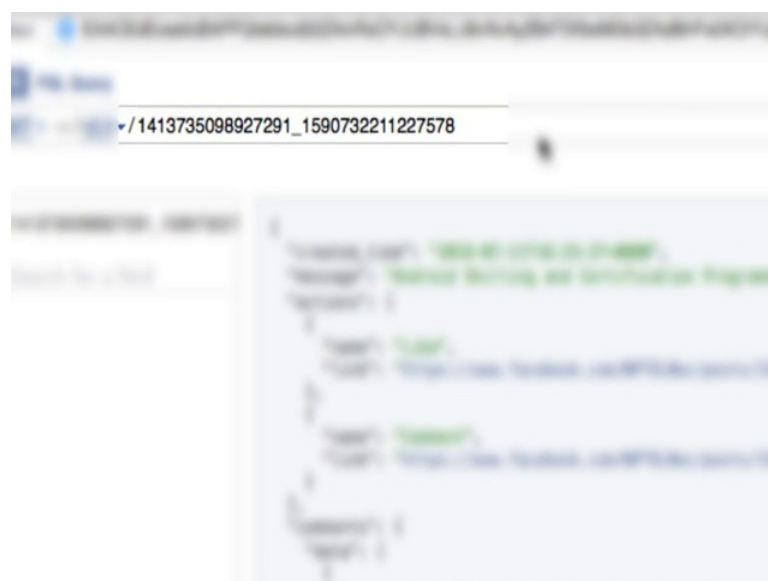
So, from the API itself, you can see that it has been shared 12 times. There is this entire list of people who have liked this post, and also the entire list of people who have commented on this post along with the common text and tags and likes on the comments etcetera.

(Refer Slide Time: 16:24)

```
d {
  "data": {
    {
      "id": "1413735098927291_1590732211227578",
      "from": {
        "name": "NPTEL",
        "category": "Education",
        "category_list": [
          {
            "id": "151676848220295",
            "name": "Education"
          }
        ],
        "id": "1413735098927291"
      },
      "message": "Android Skilling and Certification Programme launched by #GoogleIndia and available for fre",
      "picture": "https://fbcdn-photos-b-a.akamaihd.net/hphotos-ak-xfp1/v/t1.8-0/s130x130/13627126_1590732211",
      "link": "https://www.facebook.com/NPTELNoc/photos/a.1422957808005620.1073741828.1413735098927291/159073",
      "name": "Timeline Photos",
      "icon": "https://www.facebook.com/images/icons/photo.gif",
      "actions": [
        {
          "name": "Comment",
          "link": "https://www.facebook.com/1413735098927291/posts/1590732211227578"
        },
        {
          "name": "Like",
          "link": "https://www.facebook.com/1413735098927291/posts/1590732211227578"
        }
      ],
      "privacy": {
        "value": "",
        "description": "",
        "friends": "",
        "allow": "",
        "deny": ""
      }
    }
  }
}
```

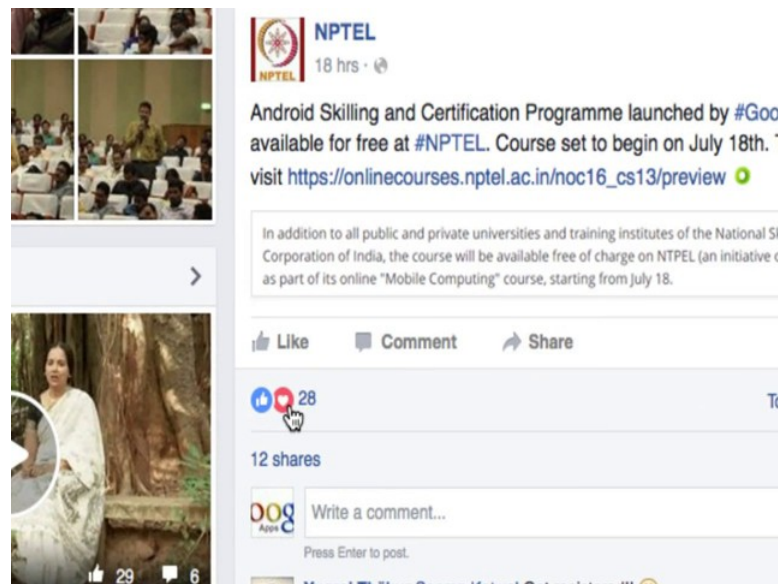
So, just like we saw that we can view individual search results by clicking on the id, we can also view individual posts by clicking on the id of the post.

(Refer Slide Time: 16:35)



Now while pages and groups etcetera have a numeric id, posts follow a slightly different format when it comes to ids. The id of each post on Facebook is of the format user id underscore post id, where user id is the id of the user who created this post. So, here the number before the underscore is the page id of the NPTEL and the number after the underscore corresponds to the post you can combine these two and you get the complete post id.

(Refer Slide Time: 17:02)



Now, Facebook recently launched reactions for post, which allows users to show reactions like happiness, anger, love, etcetera, in addition to liking a post. So, if you look again this post, there are 28 people in all who have reacted to this post, and one person has loved this post.

(Refer Slide Time: 17:21)

DEBS TOKEN: [EAPUCGUBUODAFFUWBUKJZLAFITZU TUDVSLUDVWVYLD4 IUDWNUKJLADUWIFADUUTUDYBQVJNDUGZANUEJAFITZU](#)

Graph API FQL Query

GET → /v2.0 /1413735098927291_1590732211227578

Node: 1413735098927291_159073221

Search for a field

```
{
  "created_time": "2016-07-11T16:23:37+0000",
  "message": "Android Skilling and Certification Programme launched by #GoogleIndia",
  "actions": [
    {
      "name": "Like",
      "link": "https://www.facebook.com/NPTELNoc/posts/1590732211227578"
    },
    {
      "name": "Comment",
      "link": "https://www.facebook.com/NPTELNoc/posts/1590732211227578"
    }
  ],
  "comments": {
    "data": [
      {
        "created_time": "2016-07-11T16:42:35+0000",
        "from": {
          "name": "Yugraj Thakur",
          "id": "265470150290825"
        },
        "message": "Seema Katwal Get registered!! :)",
        "can_remove": false,
        "like_count": 1,
        "message_tags": [
          {
            "id": "AR117KBBATB17"
```

Now with the version 2.6 of the API, it is possible to get these reactions on the APIs as well. Select version 2.6.

(Refer Slide Time: 17:27)

DEBS TOKEN: [EAPUCGUBUODAFFUWBUKJZLAFITZU TUDVSLUDVWVYLD4 IUDWNUKJLADUWIFADUUTUDYBQVJNDUGZANUEJAFITZU](#)

Graph API FQL Query

GET → /v2.6 /1413735098927291_1590732211227578

Node: 1413735098927291_159073221

Search for a field

```
{
  "created_time": "2016-07-11T16:23:37+0000",
  "message": "Android Skilling and Certification Programme launched by #GoogleIndia",
  "id": "1413735098927291_1590732211227578"
}
```

And type, slash reaction after the post id in the query bar.

(Refer Slide Time: 17:31)

```
_159073221 {
  "data": [
    {
      "id": "625398778981798",
      "name": "Manasvee Mittal",
      "type": "LIKE"
    },
    {
      "id": "813616272814521",
      "name": "Shoban Kalia",
      "type": "LIKE"
    },
    {
      "id": "1283998686281894",
      "name": "Mahesh Salunke",
      "type": "LIKE"
    },
    {
      "id": "18284287718411675",
      "name": "Sahebagoada Jambaladinni",
      "type": "LIKE"
    },
    {
      "id": "18288259916258726",
      "name": "Sri Balaji",
      "type": "LIKE"
    },
    {
      "id": "18153348787363568",
      "name": "Rahul Aren",
      "type": "LIKE"
    },
    {
      "id": "737486656331888",
      "name": "Basant Soni",
      "type": "LIKE"
    }
  ]
}
```

So, this returns the list of all the people who reacted to this post along with the type of reaction. You can even filter these results based on a particular reaction. So, let us say we only want the **list of** people, who loved this post, you add question mark type is equal to love in the query and press enter.

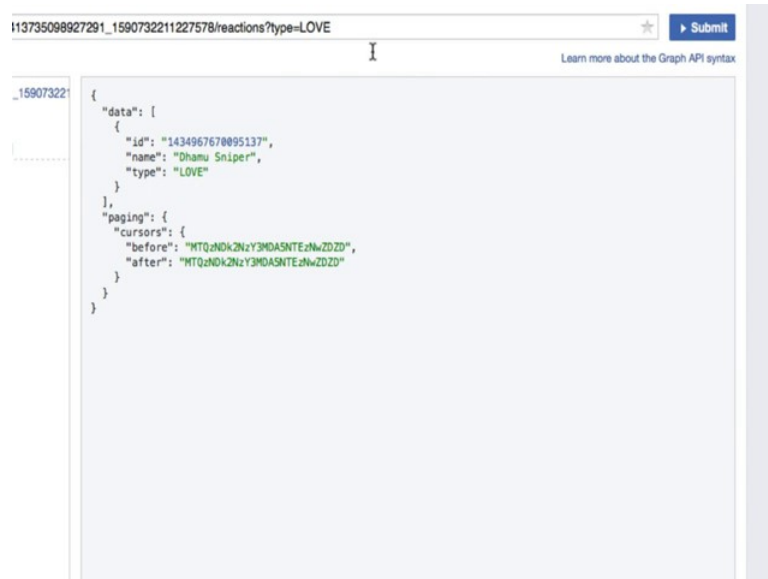
(Refer Slide Time: 17:48)

```
H3735098927291_1590732211227578/reactions?type=love
Submit
Learn more about the Graph API syntax

_159073221 {
  "error": {
    "message": "(#188) Param type must be one of {NONE, LIKE, LOVE, WOW, HAHA, SAD, ANGRY, THANKFUL}",
    "type": "OAuthException",
    "code": 188,
    "fbtrace_id": "GXX+Z6BZKT4"
  }
}
```

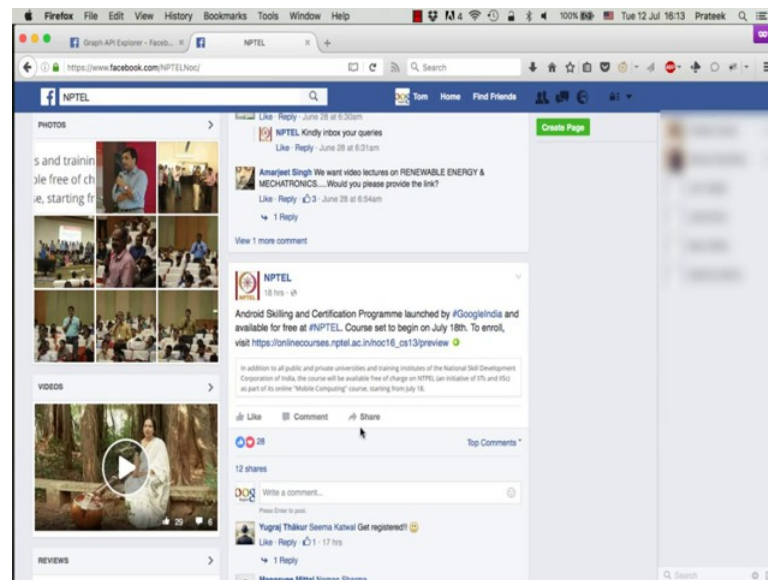
So, this looks to be case sensitive.

(Refer Slide Time: 17:51)



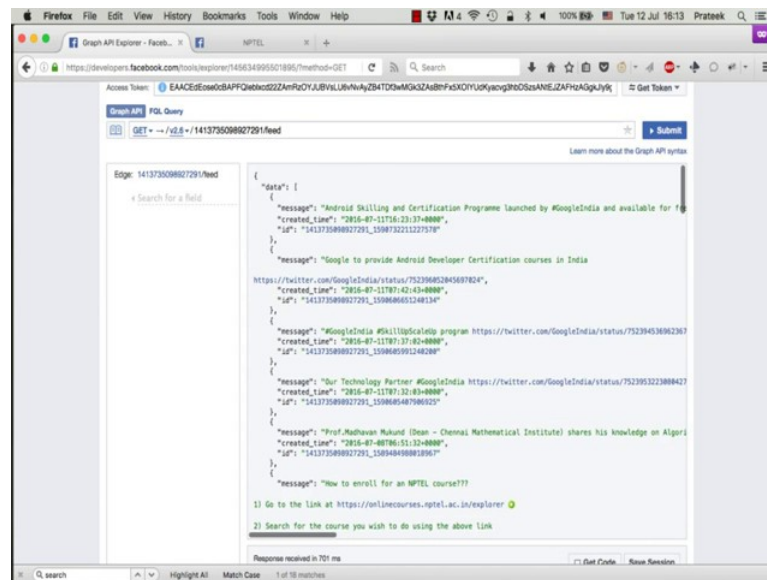
So, we type L O V E in caps and there you see; the API gives you the one person who loved this post, the name of this user is Dhamu Sniper.

(Refer Slide Time: 18:03)



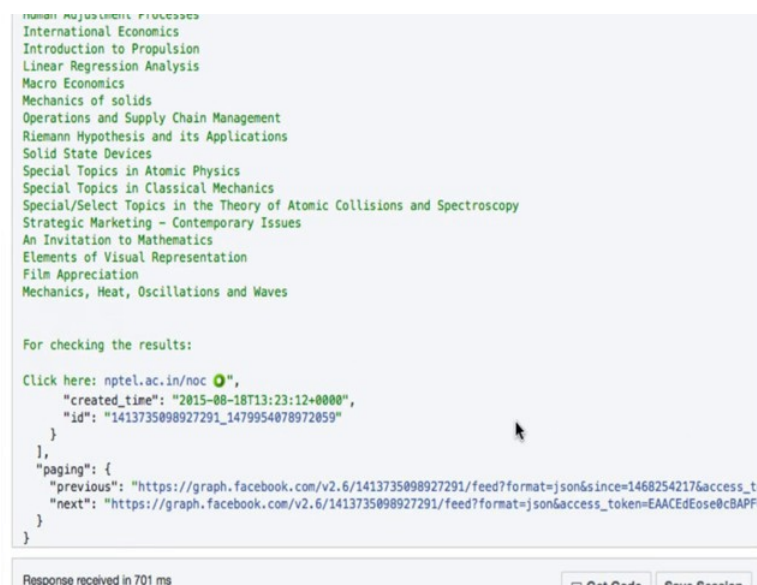
You can go back to the post on the Facebook page and verify. So, this shows the same name here too.

(Refer Slide Time: 18:19)



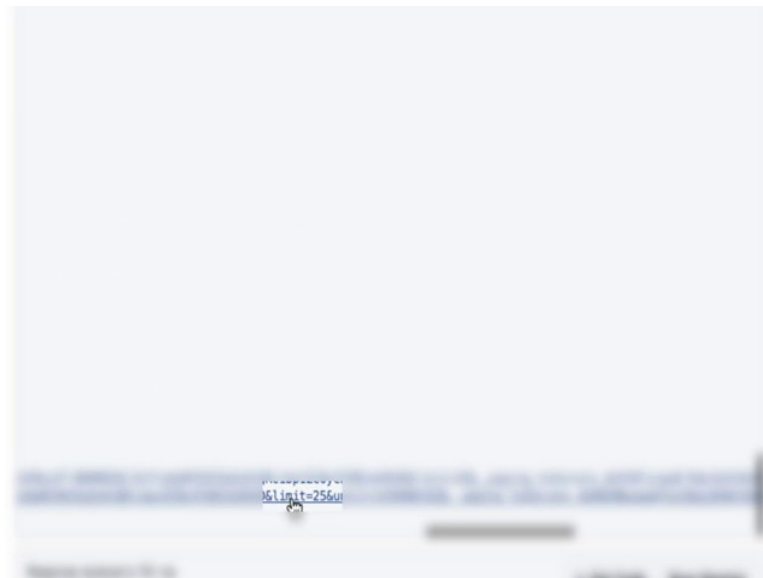
Now, there is a concept of paging in the graph API, which we have not looked at. So, whenever you make a query to the API, where the number of results is more than 25, the API returns the response in multiple pages.

(Refer Slide Time: 18:27)



If you scroll down to the very bottom of the results, you can see this paging object in the response which contains URLs to the previous and next set of results.

(Refer Slide Time: 18:37)



And if you look at this URL carefully, you will find a limit parameter which is set to 25 by default. Now when you click on this URL you get the next 25 results, which is a next 25 posts on the NPTEL page in this case.

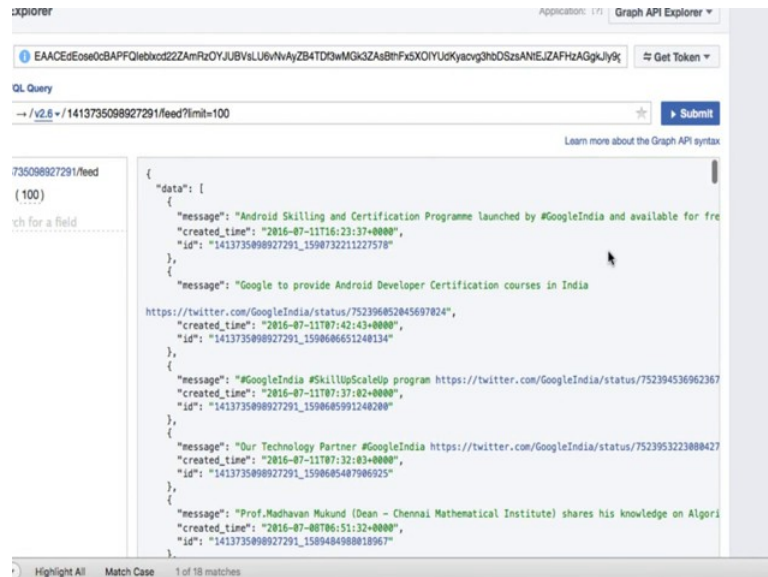
(Refer Slide Time: 18:55)



Scroll down to the bottom again and there is this next again. This URL keeps getting updated at every page and keeps on showing up again and again until all the results have been obtained. If you want to get more number of results in a single query, you can

change this limit parameter manually. So, let us change this parameter to 100 and click submit.

(Refer Slide Time: 19:10)



The screenshot shows the Graph API Explorer interface. The query bar contains the URL: `https://graph.facebook.com/v2.6/1413735098927291/feed?limit=100`. The results are displayed in a JSON format. The first few items in the 'data' array are:

```
{
  "data": [
    {
      "message": "Android Skilling and Certification Programme launched by #GoogleIndia and available for free",
      "created_time": "2016-07-11T16:23:37+0000",
      "id": "1413735098927291_159073221127578"
    },
    {
      "message": "Google to provide Android Developer Certification courses in India",
      "created_time": "2016-07-11T17:42:43+0000",
      "id": "1413735098927291_159068651248134"
    },
    {
      "message": "#GoogleIndia #SkillUpScaleUp program https://twitter.com/GoogleIndia/status/752394536962367",
      "created_time": "2016-07-11T17:37:02+0000",
      "id": "1413735098927291_1590685991248280"
    },
    {
      "message": "Our Technology Partner #GoogleIndia https://twitter.com/GoogleIndia/status/7523953223888427",
      "created_time": "2016-07-11T17:32:03+0000",
      "id": "1413735098927291_1590685487986925"
    },
    {
      "message": "Prof.Madhavan Mukund (Dean - Chennai Mathematical Institute) shares his knowledge on Algorithms",
      "created_time": "2016-07-11T18:51:32+0000",
      "id": "1413735098927291_1589484988018967"
    }
  ]
}
```

Now, if you scroll down you would notice that the number of results have increased a lot.

(Refer Slide Time: 19:22)



The screenshot shows the same Graph API Explorer interface, but the results are scrolled down to the bottom. The last item in the 'data' array is:

```
{
  "message": "Nptel India",
  "created_time": "2015-02-20T10:51:18+0000",
  "id": "1413735098927291_1415341285433339"
},
{
  "message": "Nptel India",
  "created_time": "2015-02-20T10:51:18+0000",
  "id": "1413735098927291_1415341285433339"
}

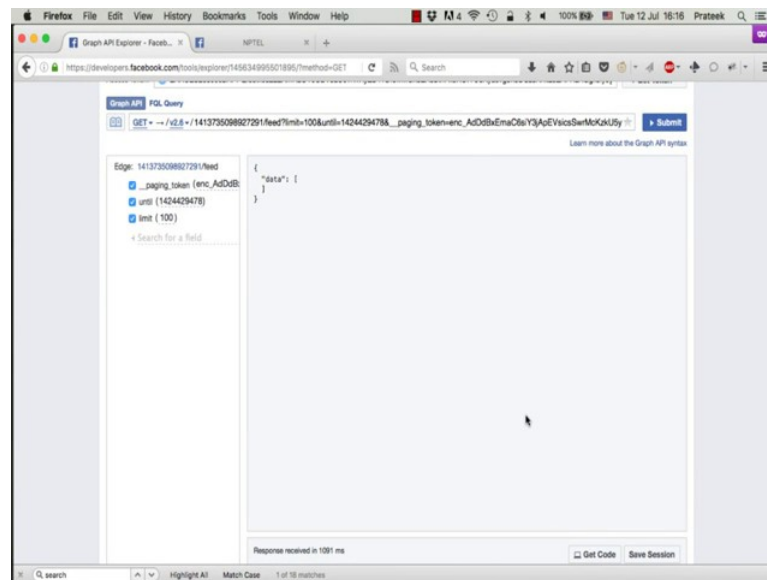
```

The 'paging' object at the bottom indicates that there are more results:

```
"paging": {
  "previous": "https://graph.facebook.com/v2.6/1413735098927291/feed?limit=100&format=json&since=1468254217",
  "next": "https://graph.facebook.com/v2.6/1413735098927291/feed?limit=100&format=json&access_token=EAAACEdE"
}
```

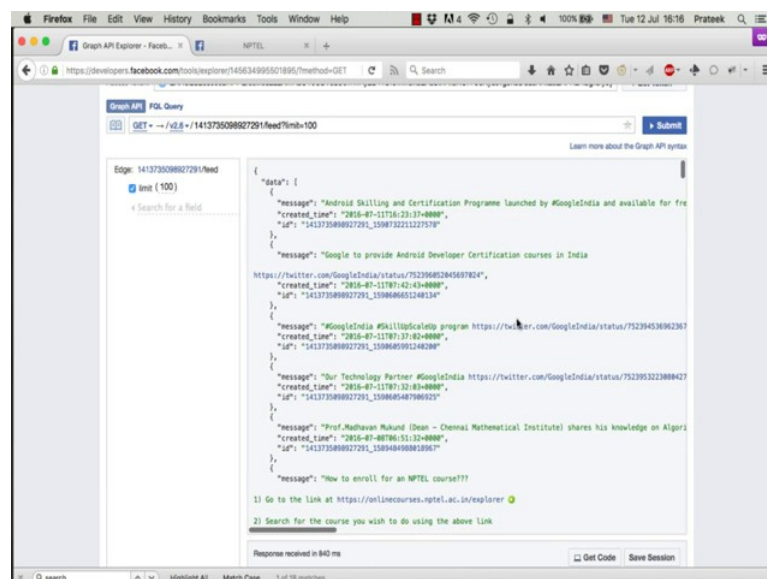
Let us scroll down to the bottom. So, there is a next again.

(Refer Slide Time: 19:26)



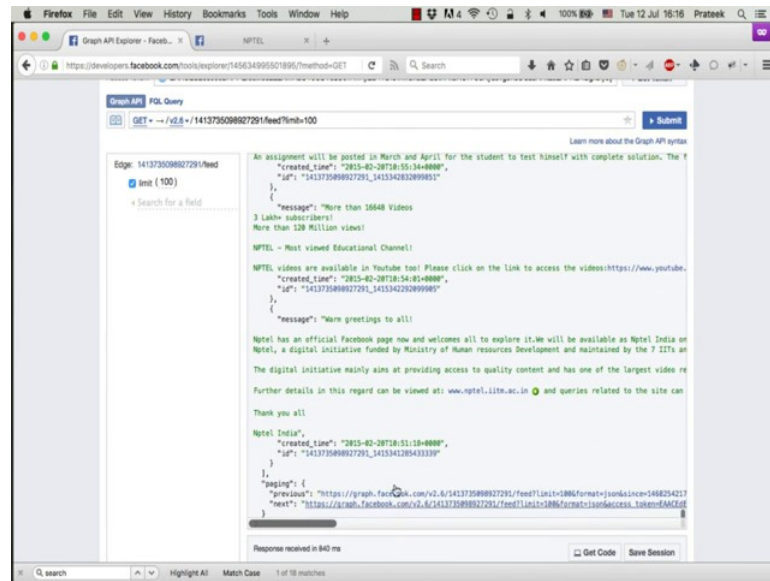
But when you click on it you see that there are no more results left the next page is blank. So, now, you know that the total number of posts in the NPTEL pages feed is 100 or less. If they were more they would have shown up in this next page.

(Refer Slide Time: 19:39)



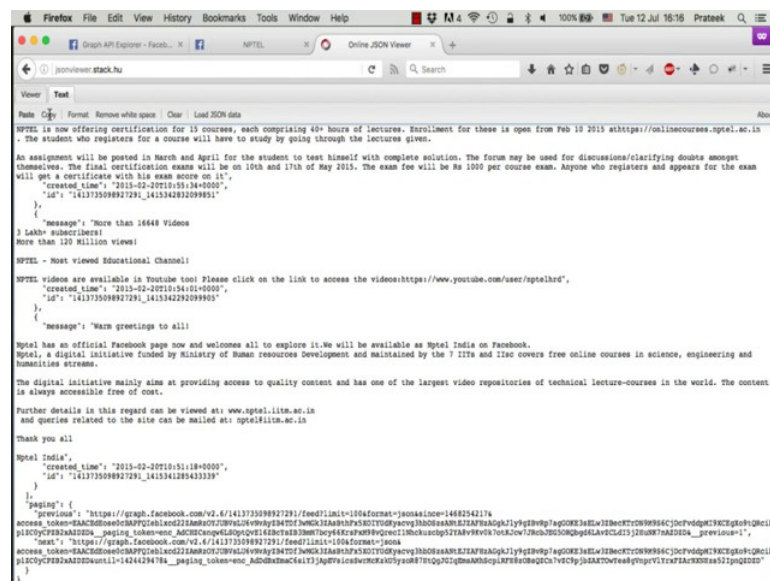
So, let us try to see exactly how many posts are there. The easiest way to do this is to copy this entire response and put it in the json viewer tool that we saw a few minutes ago.

(Refer Slide Time: 19:45)



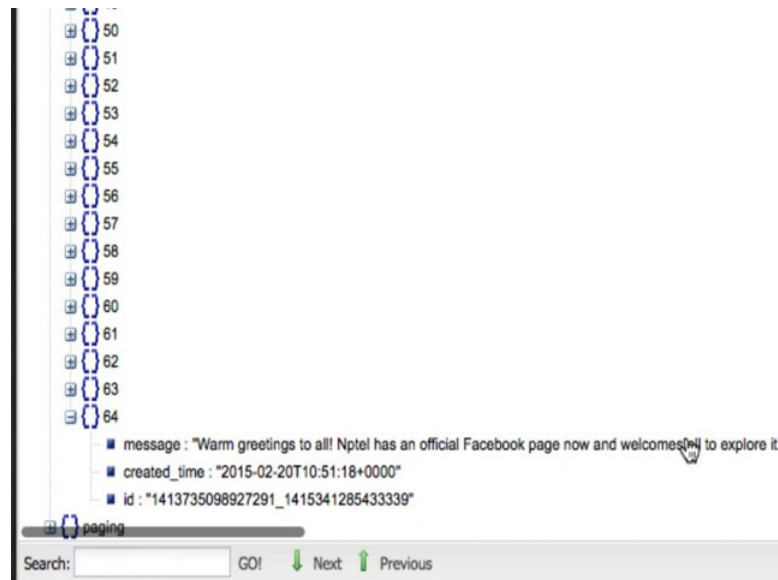
So, you select this entire response.

(Refer Slide Time: 20:12)



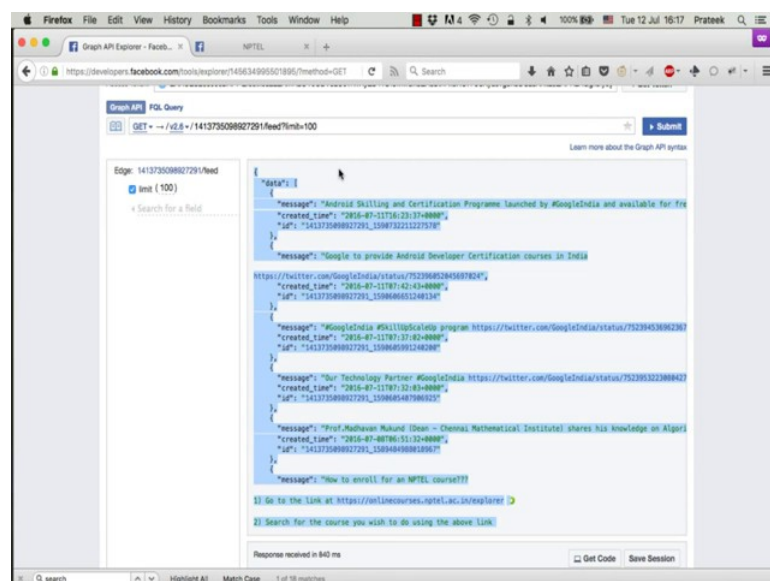
Copy and paste it in this json viewer. Go to the viewer tab.

(Refer Slide Time: 20:15)



Now expand this data object. So, it starts from 0 and goes until 64 meaning that there are total 65 public posts made by this page. Remember that the API only returns public data. So, if the page did a post where the privacy setting was not public, it will not show up in the results. So, if you expand the last post object, this was the first ever public post made by this page which was on February 28th 2015, saying warm greetings NPTEL has an official Facebook page etcetera.

(Refer Slide Time: 20:49)



But we saw earlier that this page is not a verified page. So, there is no way to really confirm that this is truly the genuine actual NPTEL page created and owned by NPTEL. So, until, now we have seen some basics of the graph API. We **have explored** some basic queries that you can make in the API; the format of the responses that the API returns etcetera.

In the next video, we will learn to do some more advanced operations with the API like creating an app, extending access tokens, collecting data programmatically using python etcetera.