

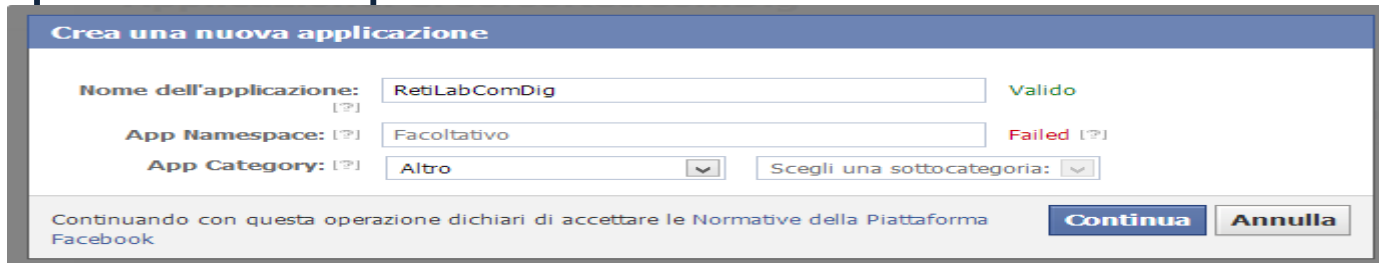
Laboratorio Reti di Calcolatori

Laurea Triennale in Comunicazione Digitale

Anno Accademico 2013/2014

Creazione App FB

1. App Dashboard <https://developers.facebook.com/apps> e cliccare su «+Crea Applicazione»
2. Compilare i campi del form



Crea una nuova applicazione


Nome dell'applicazione: Valido

App Namespace: Failed

App Category: Scegli una sottocategoria:

Continuando con questa operazione dichiari di accettare le Normative della Piattaforma Facebook

3. Completare le informazioni base dell'applicazione quali contact mail, il dominio dell'applicazione (dove applicazione verrà ospitata -> web server)



 **RetiLabComDig**

App ID: 513622565400519
App Secret: 7e4df10047b81938ada6a169a0ac8503 (reimposta)

This app is in Sandbox Mode (Only visible to Admins, Developers and Testers)

Informazioni di base

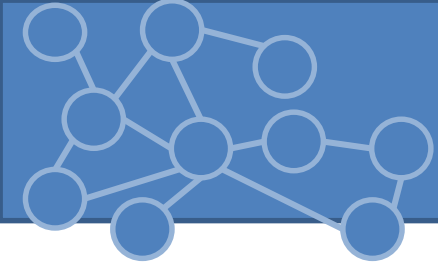
Display Name:

Namespace:

Indirizzo e-mail di contatto:

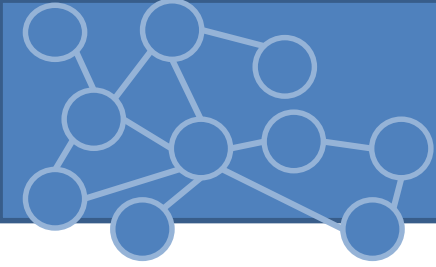
App Domains:

Modalità sandbox: Abilitato Disabilitato



Facebook Graph API

- Graph API rappresentano lo strumento principale per gestire i dati (leggere/scrivere) nel grafo sociale di FB.
 - API di tipo HTTP di basso livello usata per richiedere dati, creare post, creare check-ins o qualsiasi altro task di cui un'applicazione ha bisogno
 - Meccanismo alla base dei tool messi a disposizione da Facebook per iOS, Android, PHP e Javascript.
 - Risposte sono oggetti JSON
 - Composte da nodi, archi e campi (etichette)
 - La maggior parte richiede l'uso di un **access token**
- Permette di gestire un grafo sociale
 - Query sui nodi
 - Query connessioni tra nodi
 - Manipolazione nodi e connessione



Richiesta HTTP con metodo GET per leggere informazioni sul nodo

```
GET graph.facebook.com  
/{node-id}
```

O sull'arco

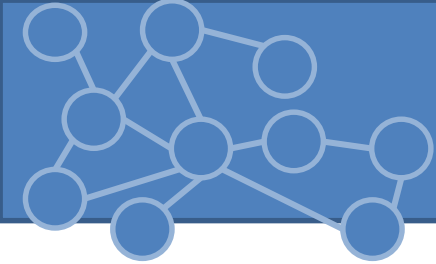
```
GET graph.facebook.com  
/{node-id}/{edge-name}
```

Posso pubblicare con metodo POST

```
POST graph.facebook.com  
/{node-id}
```

```
POST graph.facebook.com  
/{node-id}/{edge-name}
```

O eliminare con metodo DELETE



Graph API Explorer

- Graph API Explorer (GAPIEx) modo più immediato per esplorare e comprendere FB Graph API
<https://developers.facebook.com/tools/explorer>
 - Tool di livello HTTP per eseguire query, aggiungere e rimuovere risorse.
 - Utile per testare applicazione

Graph API Explorer

Home > Tools > Graph API Explorer

Application: [?] **Graph API Explorer** ▼

Access Token: × **Debug** **↔ Get Access Token**

Graph API **FQL Query**

GET ▼ **Profiling** ▼ **Submit** ▶

[Learn more](#) about new features.

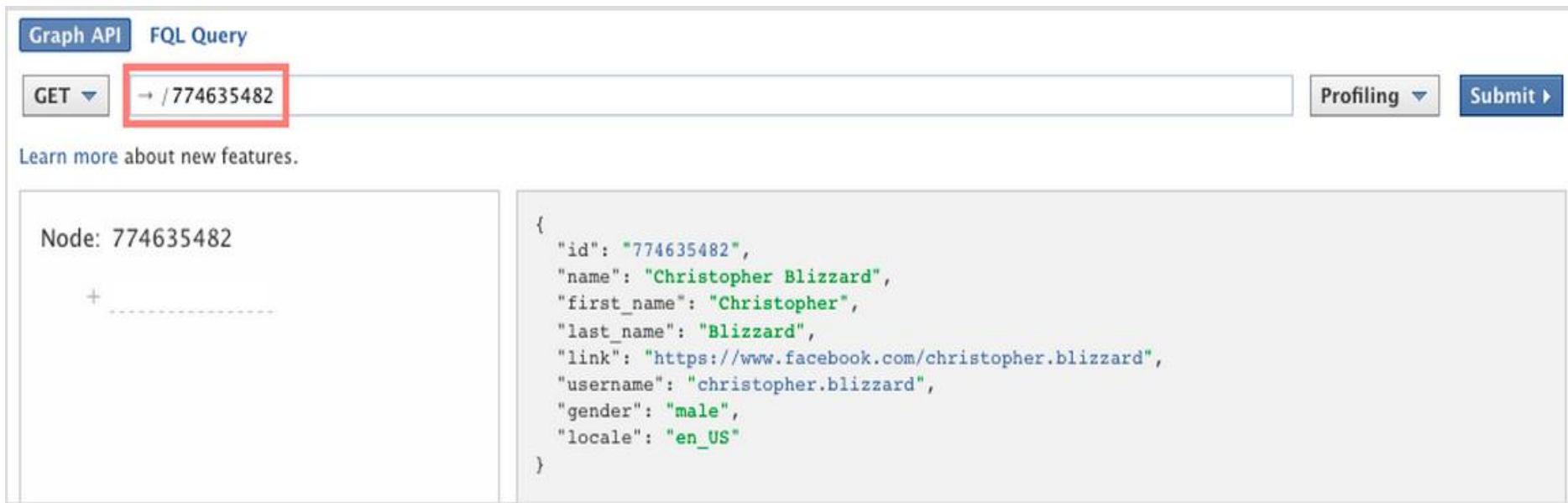
Node: 774635482

- id
- name

```
{
  "id": "774635482",
  "name": "Christopher Blizzard"
}
```

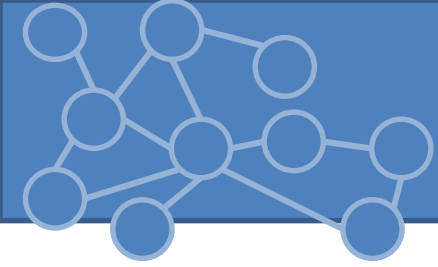
ID

- Ogni elemento in OpenGraph ha un ID
 - Il percorso per accedere alla risorsa è nella forma /<facebookId>
 - Modo per riferirmi ad un qualsiasi oggetto su cui ho permesso di accesso
 - People e Pages hanno anche un nome (persona accessibile con nome associato a ID)
 - /me



The screenshot shows the Facebook Graph API interface. At the top, there are tabs for "Graph API" and "FQL Query". Below the tabs, there is a "GET" dropdown menu and a text input field containing the path "/774635482", which is highlighted with a red box. To the right of the input field are "Profiling" and "Submit" buttons. Below the input field, there is a link that says "Learn more about new features." The main content area is split into two panels. The left panel shows "Node: 774635482" and a plus sign followed by a dashed line. The right panel shows the JSON response for the GET request:

```
{
  "id": "774635482",
  "name": "Christopher Blizzard",
  "first_name": "Christopher",
  "last_name": "Blizzard",
  "link": "https://www.facebook.com/christopher.blizzard",
  "username": "christopher.blizzard",
  "gender": "male",
  "locale": "en_US"
}
```



- Graph API basata su richieste HTTP. I metodi di HTTP si traducono in azioni sul grafo
 - GET: lettura, POST: modifica e aggiunta, DELETE: cancellazione

Graph API FQL Query

GET → /774635482?fields=id,name

[Learn more about new features.](#)

Node: 774635482

id

name

```
{  
  "id": "774635482",  
  "name": "Christopher Blizzard"  
}
```

```
GET /774635482?fields=id%2Cname HTTP/1.1  
Host: graph.facebook.com  
Connection: close
```

```
HTTP/1.1 200 OK  
Content-Type: text/javascript; charset=UTF-8  
Expires: Sat, 01 Jan 2000 00:00:00 GMT  
Date: Thu, 11 Oct 2012 17:28:02 GMT  
Connection: close  
Content-Length: 48
```

```
{"id":"774635482","name":"Christopher Blizzard"}
```

Generare un access token

- Genero un accesso token cliccando sul pulsante «Get Access Token»

Select Permissions

User Data Permissions Friends Data Permissions Extended Permissions

<input checked="" type="checkbox"/> user_about_me	<input type="checkbox"/> user_actions.books	<input type="checkbox"/> user_actions.music
<input type="checkbox"/> user_actions.news	<input type="checkbox"/> user_actions.video	<input type="checkbox"/> user_activities
<input type="checkbox"/> user_birthday	<input type="checkbox"/> user_checkins	<input type="checkbox"/> user_education_history
<input type="checkbox"/> user_events	<input type="checkbox"/> user_friends	<input type="checkbox"/> user_games_activity
<input type="checkbox"/> user_groups	<input type="checkbox"/> user_hometown	<input type="checkbox"/> user_interests
<input type="checkbox"/> user_likes	<input type="checkbox"/> user_location	<input type="checkbox"/> user_notes
<input type="checkbox"/> user_online_presence	<input type="checkbox"/> user_photo_video_tags	<input type="checkbox"/> user_photos
<input type="checkbox"/> user_questions	<input type="checkbox"/> user_relationship_details	<input type="checkbox"/> user_relationships
<input type="checkbox"/> user_religion_politics	<input type="checkbox"/> user_status	<input type="checkbox"/> user_subscriptions
<input type="checkbox"/> user_videos	<input type="checkbox"/> user_website	<input type="checkbox"/> user_work_history

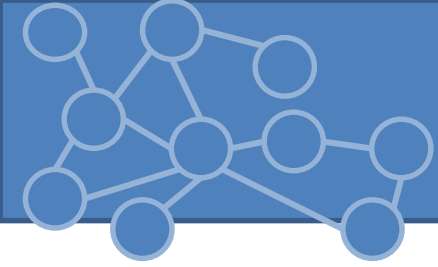
Basic permissions included by default.

Get Access Token Clear Annulla

Per ora non selezioniamo nulla

- Access token riassume i permessi affidati ad un'applicazione da un utente che possiede del contenuto all'interno del servizio.

Access Token: CAAHTixKPL8cBAICL96Yd6ipDYZBMZCZBHsBjsjodll9nGL9kgZCieUNSA43SggeWSeejR7QuCYRPAYZCq! X Debug ⇌ Get Access Token



Permessi

Finestra di dialogo dei permessi

A screenshot of a permissions dialog box. It features a list of 24 permissions arranged in three columns. The 'email' permission is selected with a blue checkmark. At the bottom, there is a text label 'Basic Permissions already included by default', a blue 'Get Access Token' button, and a grey 'Cancel' button.

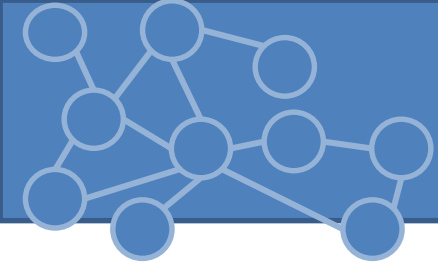
<input checked="" type="checkbox"/> email	<input type="checkbox"/> publish_actions	<input type="checkbox"/> user_about_me
<input type="checkbox"/> user_actions.music	<input type="checkbox"/> user_actions.news	<input type="checkbox"/> user_actions.video
<input type="checkbox"/> user_activities	<input type="checkbox"/> user_birthday	<input type="checkbox"/> user_education_history
<input type="checkbox"/> user_events	<input type="checkbox"/> user_games_activity	<input type="checkbox"/> user_groups
<input type="checkbox"/> user_hometown	<input type="checkbox"/> user_interests	<input type="checkbox"/> user_likes
<input type="checkbox"/> user_location	<input type="checkbox"/> user_notes	<input type="checkbox"/> user_photos
<input type="checkbox"/> user_questions	<input type="checkbox"/> user_relationship_details	<input type="checkbox"/> user_relationships
<input type="checkbox"/> user_religion_politics	<input type="checkbox"/> user_status	<input type="checkbox"/> user_subscriptions
<input type="checkbox"/> user_videos	<input type="checkbox"/> user_website	<input type="checkbox"/> user_work_history

Basic Permissions already included by default

[Get Access Token](#) [Cancel](#)

dove posso selezionare quali permessi si vogliono usare. I permessi dipendono dal tipo di dati a cui voglio accedere o cosa voglio postare.

1. Permessi richiesti al momento dell'autorizzazione
2. Permessi passati come parte della prima query di autorizzazione come insieme di stringhe.



Tipi di permessi

Ci sono 3 tipi di permessi

1. User Data Permission: dati di una persona, read-only
2. Friend's Data Permissions: dati sugli amici di una persona
3. Extended Permissions: accesso più specifico ai dati + permette la scrittura

The image shows a screenshot of the Facebook Graph API Explorer interface. On the left, there is a list of permissions with checkboxes. The permissions 'user_likes' and 'user_photos' are checked and highlighted with a red box. Below the list, it says 'Basic Permissions already included by default' and there is a 'Get Access Token' button. A large blue arrow points from this list to the right, where the main interface of the Graph API Explorer is shown. The main interface includes the Facebook logo, the title 'Graph API Explorer', and a 'Log In with Facebook' button. Below this, there is a section for the app's profile, showing a profile picture and the text '21 friends and 50,000 other people use this app'. At the bottom, there are two columns: 'ABOUT THIS APP' and 'THIS APP WILL RECEIVE:'. The 'ABOUT THIS APP' section describes the tool as a way to browse objects, manage permissions, and obtain access tokens. The 'THIS APP WILL RECEIVE' section lists the permissions granted: 'Your basic info [?]', 'Your likes', and 'Your photos'.

user_hometown user_interests user_likes
 user_location user_notes user_photos
 user_questions user_relationship_details user_relationships
 user_religion_politics user_status user_subscriptions
 user_videos user_website user_work_history

Basic Permissions already included by default [Get Access Token](#)

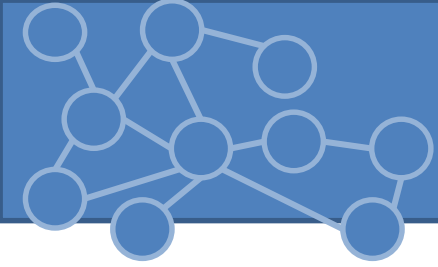
Graph API Explorer [Log In with Facebook](#)

21 friends and 50,000 other people use this app

ABOUT THIS APP
A tool to help browse objects within the Facebook Graph API, manage permissions, obtain access tokens and learn how it all works.

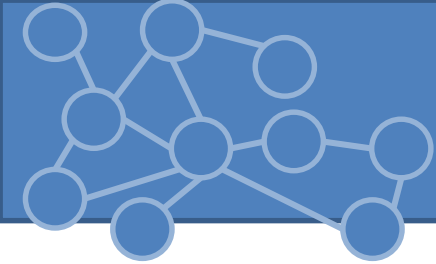
THIS APP WILL RECEIVE:

- Your basic info [?]
- Your likes
- Your photos



Connessioni

- Oggetti connessi tramite relazioni = CONNECTIONS
- Posso estrarre le connessioni in cui un oggetto è coinvolto
https://graph.facebook.com/ID/CONNECTION_TYPE
- CONNECTION_TYPE: friends, home, news feed, likes, movies, profile feed, books, notes, permissions, photo tag, photo album, video tag, video upload, events, groups, checkins, oggetti con location



Leggere i dati

- Graph API permette lettura delle proprietà e delle connessioni di un oggetto
- SELEZIONE
 - Per default vengono restituite la maggior parte delle proprietà di un oggetto
 - Posso scegliere quali campi devono essere restituiti usando il parametro **'fields'** nella query
<https://graph.facebook.com/ID?fields=id,name,picture>
 - Posso richiedere più oggetti in una singola query usando il parametro **'ids'**
<https://graph.facebook.com?ids=ID1,ID2,...,IDN>

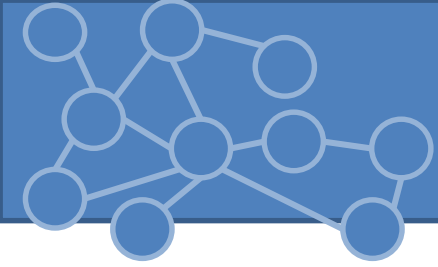
- Richieste Multiple in una singola chiamata

```
GET graph.facebook.com
/{node-id}?
  fields={first-level}.fields({second-level})
```

First level: campi o connessioni (arco) del nodo padre separati da virgola
Second level: : campi o connessioni (arco) del nodo first level separati da virgola

- Uso **.limit(N)** per indicare quanti oggetti voglio

```
GET graph.facebook.com
/me?
  fields=albums.limit(5).fields(name, photos.limit(2)),posts.limit(5)
```

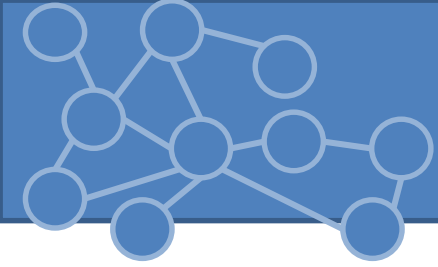


IMPAGINAZIONE

Quando eseguo una query non ricevo tutti i risultati, ma vengono suddivisi in pagine. **Cursor pagination** è un metodo di suddivisione in pagine della risposta.

Un cursore è una stringa casuale usata come marcatore in una lista di dati.

```
{
  "data": [
    ... Endpoint data is here
  ],
  "paging": {
    "cursors": {
      "after": "MTAxNTExOTQ1MjAwNzI5NDE=",
      "before": "NDMyNzQyODI3OTQw"
    },
    "previous": "https://graph.facebook.com/me/albums?limit=25&before=NDMyNzQyODI3OTQw"
    "next": "https://graph.facebook.com/me/albums?limit=25&after=MTAxNTExOTQ1MjAwNzI5NDE="
  }
}
```

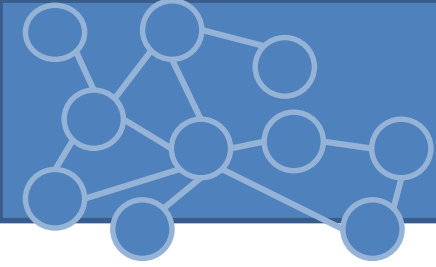


Supporta i parametri

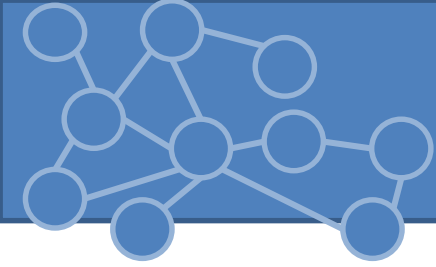
- Before: cursore che punta all'inizio dei dati della pagina inviata.
- After: cursore che punta alla fine dei dati della pagina inviata.
- Previous: restituisce la pagina precedente
- Next: restituisce la prossima pagina
- Limit: massimo numero di oggetti restituiti nella pagina

Quando le pagine finiscono, l'ultima pagine non contiene next, tuttavia posso sfruttare after per verificare se informazioni sono state aggiornate

```
GET https://graph.facebook.com/12345/comments?limit=25&after=abcd1234
```



- Alcuni parametri per l'impaginazione e filtratura dei risultati di una query
 - Limit: quanti elementi mi vengono restituiti
`graph.facebook.com/ID/likes?limit=4`
 - Offset: spostamento dei risultati
`graph.facebook.com/ID/likes?offset=3`
 - Until: tutti i risultati fino ad una data specificata
`graph.facebook.com/ID/likes?until=yesterday`
 - Since: tutti i risultati a partire dalla data specificata
`graph.facebook.com/ID/likes?since=today`
 - Il formato della data è quello specificato dalla ISO 8601
YYYY-MM-DDThh:mm:ss.sTZD. I valori accettati sono unixtimestamp e tutti quelli riconosciuti dalla funzione `strtotime` di PHP



Ricerca

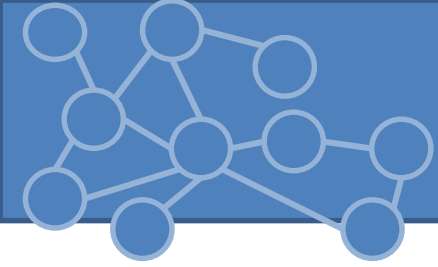
- Si possono eseguire ricerche sugli oggetti pubblici con <https://graph.facebook.com/search>
- Formato generale

https://graph.facebook.com/search?q=QUERY&type=OBJECT_TYPE

- OBJECT_TYPE: post pubblici, page, event, group, place (*center* e *distance*), checkin (non accetta il parametro q), people, application
 - Oggetti geo-localizzati: type:location e parametri center e distance
 - Oggetti relativi ad un luogo: type=location e place=ID
- Posso restringere i risultati ad alcuni campi usando il parametro 'fields'
- Posso cercare nella home dell'utente aggiungendo l'argomento 'q' alla connessione home. Dati fino a 2 settimane.
 - <https://graph.facebook.com/me/home?q=QUERY>

```
GET graph.facebook.com
/search?
q=coffee&
type=place&
center=37.76,-122.427&
distance=1000
```

```
GET graph.facebook.com
/search?
type=location&
center=37.76,-122.427&
distance=1000
```

Scrivere i dati

- Posso pubblicare sul grafo di FB usando una richiesta HTTP POST all'URL di un oggetto che accetta uno stream
 - Passo l'access_token e gli argomenti necessari

Method	Description	Arguments
<code>/PROFILE_ID/feed</code>	Publish a new post on the given profile's feed/wall	<code>message</code> , <code>picture</code> , <code>link</code> , <code>name</code> , <code>caption</code> , <code>description</code> , <code>source</code> , <code>place</code> , <code>tags</code>
<code>/OBJECT_ID/comments</code>	Comment on the given object (if it has a <code>/comments</code> connection)	<code>message</code>
<code>/OBJECT_ID/likes</code>	Like the given object (if it has a <code>/likes</code> connection)	<i>none</i>
<code>/PROFILE_ID/notes</code>	Publish a note on the given profile	<code>message</code> , <code>subject</code>
<code>/PROFILE_ID/links</code>	Publish a link on the given profile	<code>link</code> , <code>message</code> , <code>picture</code> , <code>name</code> , <code>caption</code> , <code>description</code>
<code>/PROFILE_ID/events</code>	Create an event	<code>name</code> , <code>start_time</code> , <code>end_time</code>
<code>/EVENT_ID/attending</code>	RSVP "attending" to the given event	<i>none</i>
<code>/EVENT_ID/maybe</code>	RSVP "maybe" to the given event	<i>none</i>
<code>/EVENT_ID/declined</code>	RSVP "declined" to the given event	<i>none</i>
<code>/PROFILE_ID/albums</code>	Create an album	<code>name</code> , <code>message</code>
<code>/ALBUM_ID/photos</code>	Upload a photo to an album	<code>message</code> , <code>source</code> , <code>place</code> (<i>multipart/form-data</i>)
<code>/PROFILE_ID/checkins</code>	Create a checkin at a location represented by a Page	<code>coordinates</code> , <code>place</code> , <code>message</code> , <code>tags</code>



Esercizi

- Usando Graph API Explorer (GAPIEx) si salvi la risposta alla richiesta 'https://graph.facebook.com/platform' in un file di testo. Una volta salvato si scriva un prg che decodifichi il file e che mostri tutti i campi coi relativi valori.
- Si ripeta l'ultima parte dell'esercizio 6, anziché utilizzare il file ci si connetta all'URL specificato e si utilizzi la risposta ad una richiesta di tipo GET come punto di partenza per la decodifica
- In GAPIEx fatevi restituire i film che vi piacciono e salvateli su un file 'movie.txt'
- Partendo da 'movie.txt' costruite una statistica che vi restituisce il vostro genere preferito
- In GAPI si cerchino tutti gli eventi che nel nome hanno 'happy' creati a partire da ieri
- Si esegua la stessa operazione usando la classe HttpURLConnection
- Sul proprio wall si posti un messaggio e poi lo si commenti
- Si scriva una breve nota
In 12 e 13 si utilizza GAPIEx