

NOMBRE: Nadia Corina Garcia Orozco

GITHUB REPOSITORY : [ <https://github.com/ncgo/ClaseWeb/tree/master/Lab7> ]

GITHUB PAGE URL: [ <https://ncgo.github.io/ClaseWeb/Lab7/index.html> ]

## AJAX

En este laboratorio utilizarás el API de GIPHY para desarrollar una página dinámica con gifs de tu elección. Adicionalmente deberás utilizar jQuery para crear elementos HTML.

### Antes de Comenzar

- Familiarízate con el API de GIPHY [Giphy API](#).
- Asegúrate de leer acerca de los siguientes parámetros:
  - **q**
  - **limit**
  - **rating**
- Como la gran mayoría de los APIS, es necesario registrar una cuenta y obtener un API Key.
- Asegúrate de cambiar el protocolo de query de **http** a **https** o la aplicación no funcionará correctamente cuando la subas a Github Pages.

### **Instrucciones**

1. Antes de comenzar, necesitas crear un arreglo de Strings, cada elemento relacionado a un tema de tu preferencia (pueden ser de varios temas). Guárdalo en una variable llamada: **temas**
  - En el ejemplo se utilizaron animales como el tema principal, pero tu puedes hacer la lista a tu elección.
2. De manera inicial, tu archivo "index.html" deberá invocar "script.js" el cual utilizará un loop con el arreglo previamente creado y generará un botón por cada String. (Utiliza jQuery).
3. Cuando el usuario haga click en el botón, la página deberá realizar una consulta al API de GIPHY, tomar 10 elementos estáticos (no animados) y desplegar cada uno de ellos en la página. (Deberás incluir la lógica correspondiente al "click" para cada botón. (Recuerda el tema de delegación de eventos). Abajo o arriba de cada gif, despliega su rating (PG, G, etc.)

### **[HINT]**

Para poder guardar información temporal en cualquier elemento html, en este caso en el tag <img> puedes utilizar "data-attributes" (ver referencia #1). Tener "data-attributes" te permitirá almacenar información temporal. La idea es que generes cada botón con al menos los siguientes atributos:

- Imagen sin movimiento
- Imagen con movimiento
- Variable que indique el estatus de la imagen.

De la consulta al API necesitarás los siguientes datos:

- rating
- images.fixed\_height.url
- images.fixed\_height\_still.url

4. Cuando el usuario haga click en alguna de las imágenes, el gif debería de reproducir la animación correspondiente. Si el usuario vuelve a hacer click, la animación deberá detenerse.

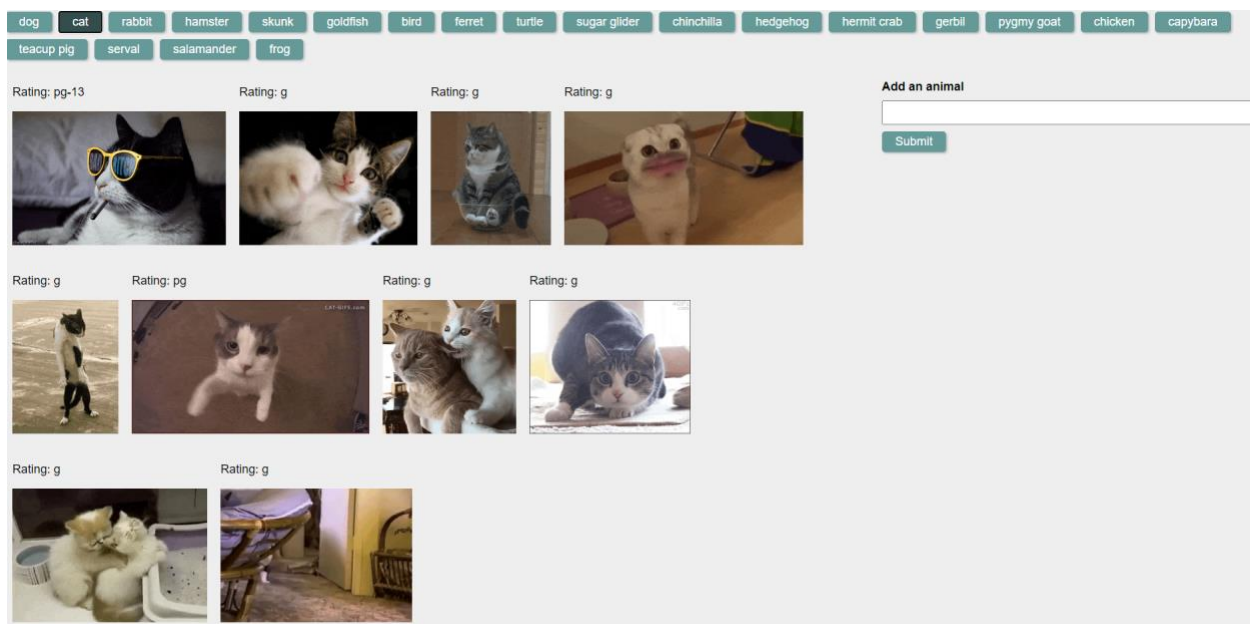
[HINT]

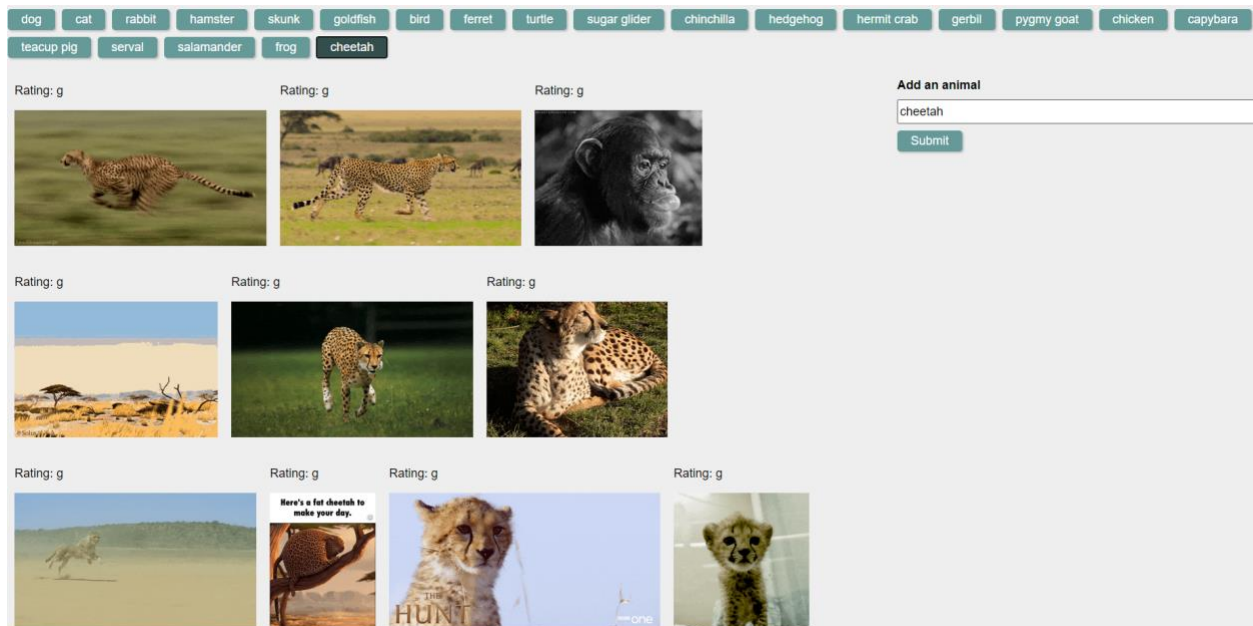
Esto lo puedes lograr comparando el estatus actual de la imagen (si se encuentra en movimiento ó no) así como el método `.attr(propiedad, valor)` de jQuery.

5. Agrega un formulario que tome el valor ingresado por el usuario y lo agrega al arreglo de **temas**. Crea una función que tome cada elemento del arreglo y re-genere los botones en la página.

[HINT]

Recuerda agregar un `“preventDefault()”` para evitar que la página se cargue nuevamente al dar click en el botón `“Submit”`.





## REFERENCIAS:

- #1 <https://www.abeautifulsite.net/working-with-html5-data-attributes>

Una vez que termines el desarrollo deberás de publicar este laboratorio en un nuevo repositorio (TIENE QUE SER DIFERENTE A LOS REPOSITORIOS DE LOS LABORATORIOS ANTERIORES DE LO CONTRARIO SE TE DESCONTARÁN 15 PUNTOS). Convierte este repositorio en un Github pages para poder ver su versión en vivo y sube a CANVAS ambas urls, una del github page y otra del repositorio.

**“If you try and Fail, Congratulations.  
Most People won’t even try”**