

Taller de Git, GitHub y GitFlow

Ejercicio 2

21 de Febrero de 2017

Autores: Alfredo González Gaviña,

Walter Alejandro Moreno Ramírez y María Susana Ávila García.

El objetivo de este ejercicio es realizar una pagina de web sencilla, utilizando HTML 5, CSS 3 y JavaScript con la finalidad de verificar como es el manejo de la herramienta GitKraken.

Al final de esta práctica el estudiante será capaz de crear un nuevo repositorio en GitKraken, realizar las funciones básicas de Git dentro de dicha herramienta.

1.- Configuraciones Iniciales del repositorio

En esta parte se muestran las configuraciones iniciales de git, así como la creación del primer repositorios y los comandos más básicos (add, status, commit).

1.1 Inicialización de un nuevo repositorio

Para realizar la inicialización de un nuevo repositorio de git se usa el comando init. La manera de inicializar un repositorio con el nombre de “paginaWeb” es:

```
git init paginaWeb
```

La inicialización exitosa del repositorio arroja un mensaje como este, donde se indica que un repositorio vacío se ha inicializado junto al directorio donde se realizó esta acción:

```
Initialized empty Git repository in /home/walter/MEGAsync/DevProjects/tallerGit/.git/
```

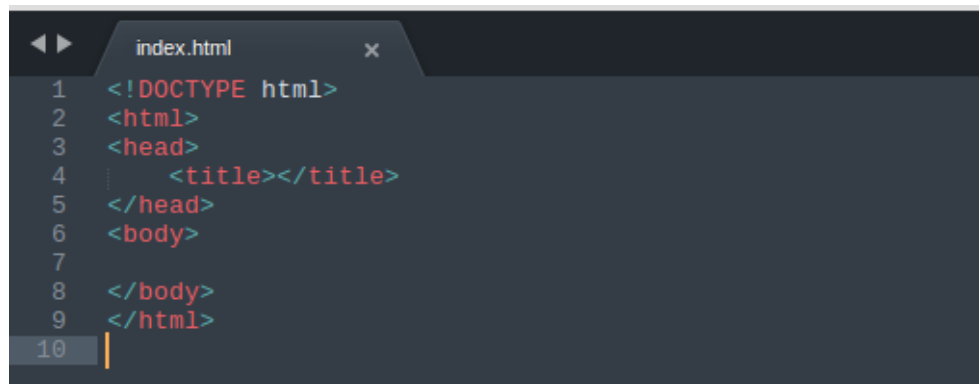
Nota: Al inicializar un nuevo repositorio, Git crea archivos para su control, estos archivos no deben ser modificados ni eliminados.

El siguiente paso es agregar nuestro nombre de usuario e email de nuestra cuenta de GitHub utilizando los siguientes comandos:

```
git config --global user.username "tu usuario de GitHub"  
git config --global user.email "tu contraseña de GitHub"
```

2.- Creación del archivo index.html

Utilizando el editor de texto que prefieras, crea un archivo que se llame `index.html` y dentro, escribe el prototipo básico para cualquier página web, mostrado en la Figura 1. Este archivo lo guardamos en la carpeta donde inicializamos el repositorio.

A screenshot of a code editor with a dark theme. The editor shows a file named 'index.html' with a tab at the top. The code is as follows:

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title></title>
5 </head>
6 <body>
7
8 </body>
9 </html>
10
```

Figura 1. Prototipo básico para cualquier página web.

Para poder ver el estado de los archivos dentro del repositorio utilizamos el comando

```
git status
```

Lo que nos mostrará el archivo `index.html` de reciente creación.

Para poder agregar el archivo y sus cambios agregados a la pila de preparación utilizamos el comando:

```
git add index.html
```

Nota: si queremos agregar los cambios realizados en un sólo archivo, debemos especificarlo con su nombre y extensión. De otra manera, si lo que deseamos es agregar todos los cambios realizados simplemente agregamos un `*` que indica que agregaremos todos los archivos dentro del directorio.

Confirmamos que el archivo `index.html` con el esqueleto de la página web lo agregaremos al repositorio con el siguiente comando:

```
git commit
```

3.- Agregando información básica a la página

El siguiente paso es agregar un título, un banner principal, una imagen y texto. El título lo agregaremos dentro del `<head>` utilizando la etiqueta `<title>`. Agregaremos como título “Taller de Git” como en la Figura 2.

```
3 <head>
4   <title> Taller de Git </title>
5 </head>
```

Figura 2. Agregando el título a la página.

Dentro la etiqueta `<header>` agregamos la siguiente información:

Utilizando la etiqueta `<h1>` agregamos: “Taller de Git, GitHub y GitFlow”

Y utilizando la etiqueta `<h4>` agregamos: “Se llevará a cabo los días 21 y 28 de Febrero, 7 y 14 de Marzo”, tal como se muestra en la Figura 3.

```
<body>
  <header>
    <h1> Taller de Git, GitHub y GitFlow </h1>
    <h4> Se llevará a cabo los días 21 y 28 de Febrero, 7 y 14 de Marzo </h4>
  </header>
```

Figura 3. Agregando el header a la página.

Después de agregar el `<header>`, creamos un contenedor `<div>` y, utilizando la etiqueta de para párrafos `<p>`, dentro agregamos el siguiente texto:

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Puedes utilizar la etiqueta `
` (no es necesario cerrar la etiqueta) para hacer un salto de línea y dar el formato que desees.

Como elemento final, agregaremos una imagen a nuestra página, utilizaremos la etiqueta `` utilizando su atributo “src”. El resultado final del código se muestra en la Figura 4.

```

<!DOCTYPE html>
<html>
<head>
  <title> Taller de Git </title>
</head>
<body>
  <header>
    <h1> Taller de Git, GitHub y GitFlow </h1>
    <h4> Se llevará a cabo los días 21 y 28 de Febrero, 7 y 14 de Marzo </h4>
  </header>

  <div>
    <p>
      Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod
      tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,
      quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo
      consequat. <br><br>

      Duis aute irure dolor in reprehenderit in voluptate velit esse
      cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non
      proident, sunt in culpa qui officia deserunt mollit anim id est laborum.
      Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod
      tempor incididunt ut labore et dolore magna aliqua. <br><br>

      Ut enim ad minim veniam,
      quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo
      consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse
      cillum dolore eu fugiat nulla pariatur.

      Excepteur sint occaecat cupidatat non
      proident, sunt in culpa qui officia deserunt mollit anim id est laborum.
    </p>

    
  </div>
</body>
</html>

```

Figura 4. Código de la página web con sus modificaciones.

Y el resultado de estos cambios debe verse como la Figura 5.

Taller de Git, GitHub y GitFlow

Se llevará a cabo los días 21 y 28 de Febrero, 7 y 14 de Marzo

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.



Figura 5. Maquetación de la página con elementos de texto e imágenes.

De igual manera, agregamos todos los cambios a la pila de preparación con el comando:

```
git add *
```

Y realizamos un segundo commit para guardar los cambios en el repositorio en una nueva versión con:

```
git commit
```

4. Agregar estilos a la página web

El siguiente paso es agregar colores, redimensionar el tamaño de la imagen y orden de los elementos con CSS 3.

En la misma carpeta donde tenemos el index.html, creamos el archivo indexStyle.css. Dentro de este archivo vamos a definir todos elementos de estilos. Pero antes, a cada etiqueta donde tengamos a los elementos debemos agregarles la propiedad class tal como se muestra en la Figura 6.

```
<div class="content">
<p class="txtContent">

```

Figura 6. Se les agrega una clase para cada tipo diferente de etiqueta.

Utilizaremos estos selectores de clase para aplicar los estilos correspondientes.

Cambiaremos el color de fondo de la página a un gris claro modificando la etiqueta <body> directamente de acuerdo a la Figura 7.

```
body{
    background-color: rgb(210, 210, 210);
}
```

Figura 7. Definición del color de fondo de toda la página web, mediante el selector a la etiqueta *body*.

Para el *header* cambiaremos distintos aspectos: color de fondo, padding o espaciado interno, la alineación del texto y el color del texto de acuerdo a la Figura 8.

```
header{
    background-color: rgb(120,0,0);
    padding: 15px;
    text-align: center;
    color: white;
}
```

Figura 8. Modificaciones del color de fondo, espaciado interno y la alineación y color del texto para el *header*.

Para los directores “content”, “txtContent” e “imgPoster”, agregamos los siguientes estilos mostrados en la Figura 9.

```

.imgPoster{
  width: 25%;
  height: 25%;
}

.txtContent{
  margin-right: 15px;
  padding: 20px;
  background-color: white;
  width: 70%;
  height: 100%;
  border-left: 6px solid rgb(120,0,0);
}

.content{
  width: 100%;
  display: inline-flex;
  text-align: left;
}

```

Figura 9. Modificaciones de distintos parámetros para los directores *content*, *txtContent* e *imgPoster*.

El resultado final de la página con los estilos agregados se muestra en la Figura 10.



De la misma manera, agregamos todos los cambios a la pila de preparación con el comando:

```
git add *
```

Y confirmación los cambios para que sean subidos al repositorio en una nueva versión con:

```
git commit
```

Al finalizar este ejercicio, contamos con dos archivos dentro de nuestro repositorio, la página principal *index.html* y la hoja de estilos *indexStyle.css*. De la misma manera podemos seguir agregando, modificando o eliminando archivos a nuestro proyecto y Git nos permitirá crear diferentes versiones de nuestra página web de una manera muy sencilla.