**Web Programming-A practical Approach(22CS352)**

**Lab manual**

**Termwork 1:**

**Problem definition: Design web page structure using Figma**

**Theory :**

**Figma** is a [collaborative](https://en.wikipedia.org/wiki/Collaborative_software) [web application](https://en.wikipedia.org/wiki/Web_application) for [interface design](https://en.wikipedia.org/wiki/Interface_design), with additional offline features enabled by desktop applications for [macOS](https://en.wikipedia.org/wiki/MacOS) and [Windows](https://en.wikipedia.org/wiki/Microsoft_Windows). The feature set of Figma focuses on [user interface](https://en.wikipedia.org/wiki/User_interface_design) and [user experience](https://en.wikipedia.org/wiki/User_experience_design) design, with an emphasis on real-time collaboration,[[1]](https://en.wikipedia.org/wiki/Figma_(software)#cite_note-1) utilising a variety of [vector graphics editor](https://en.wikipedia.org/wiki/Vector_graphics_editor) and [prototyping](https://en.wikipedia.org/wiki/Software_prototyping) tools. The Figma mobile app for [Android](https://en.wikipedia.org/wiki/Android_(operating_system)) and [iOS](https://en.wikipedia.org/wiki/IOS) allows viewing and interacting with Figma prototypes in real-time on mobile and tablet devices.

Steps to be followed for design using Figma tool:

**Step 1:** Setting the page template

* How to Create a New Page?
* What are the Layers in a Page Template?

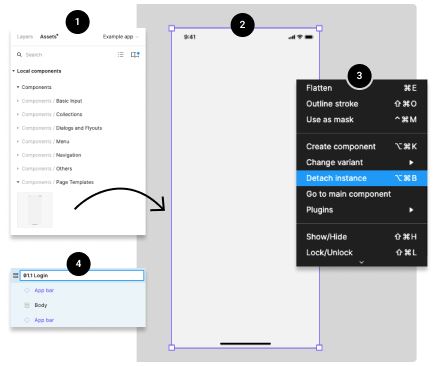
**Step 2:** Creating a form

* How to Add a Text Box?
* How to Change the Icon of a Text Box?
* What Are the Layers in a Text Box?
* How to Add a Password Box?
* How to Change the Icon of a Password Box?
* What Are the Layers in a Password Box?
* How to Add a Button?

**Step 3:** Adjusting the layout

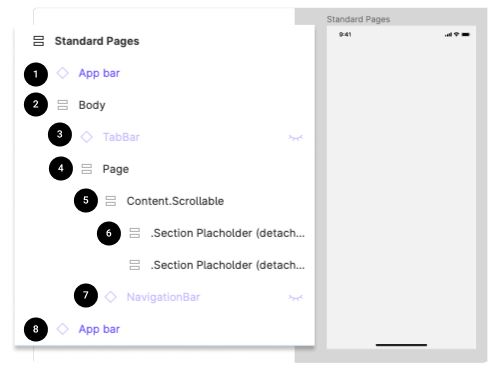
* How to Edit the Spacing Between the Elements?
* How to Edit the Padding Around the Items?
* How to Make the Content Centered Vertically?

Step 1: Setting the page template



### How to Create a New Page?

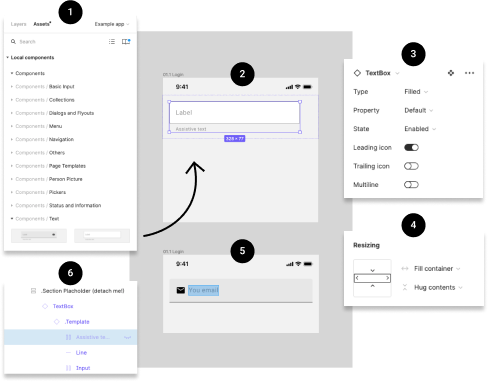
1. **Drag the Page Template** from the Page Templates components library (left side of your screen) into your Figma project.
2. **Select the page** by clicking on it. You should see a purple outline around the page.
3. **Detach the page component** from his instance by right-clicking directly on the page or the page's layers), and choosing the Detach Instance option
4. **Rename the page** by double-clicking the layer's name directly from the layers panel (right side of your screen).



### What Are the Layers in a Page Template?

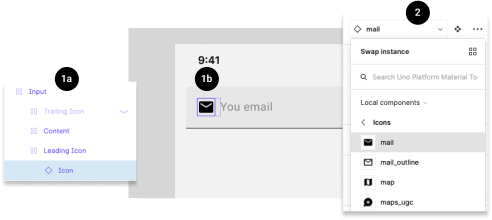
1. **App bar:** iOS or Android bottom app bar. Not part of the actual app, but it's only for presentation purposes.
2. **Body:** Contain the Bottom Navigation Bar component and the Page layers.
3. **Top App Bar:** The sticky top bar that contains the commands and page title. If this layer is invisible, this component is not relevant to the actual page, and it could be deleted from the layers.
4. **Page:** Contain all the Content. Scrollable layers and the Top App Bar component.
5. **Content.Scrollable:** Contain all the scrollable content.
6. **Section Placeholder (2):** We added 2 section placeholder to help you insert content faster into your design
7. **Navigation Bar:** The sticky bottom navigation bar. If this layer is invisible, this component is not relevant to the actual page, and it could be deleted from the layers.
8. **App bar:** iOS or Android top app bar. Not part of the actual app, but it's only for presentation purposes.

## Step 2: Creating a form



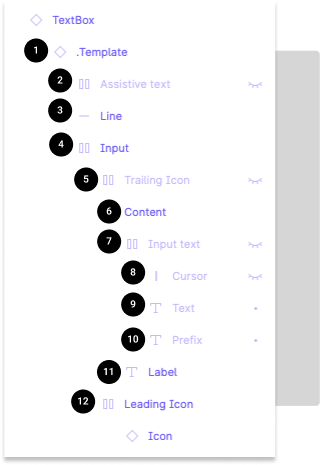
### How to Add a Text Box?

1. **Drag the Text Box** from the Text components library (left side of your screen) into your Figma project.
2. **Select the Text Box** by clicking on it. You should see a purple outline around the component.
3. **Customize the look and content** of the Text Box with the options you need from the variants control panel (right side of your screen).
4. **Adjust the width of the Text Box** by changing the value from "fixed width" to "fill container" in the Resizing panel (right side of your screen)
5. **Edit the text of the Text Box.** To do so, you could click on the label until the text becomes highlighted or select the label layer from the layers panel (left side of the screen). Once selected, use your keyboard to enter new text.
6. **Make the Assistive label invisible** if you are not using it.



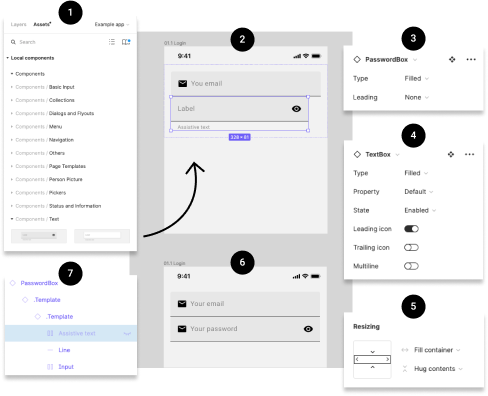
### How to Change the Icon of a Text Box?

1. **Select the icon** from a Text Box by clicking directly on the icon multiple times until the icon becomes highlighted (1b) or by selecting the Icon layer from the layer panel (left side of your screen - 1a).
2. **Choose a new icon** from the drop-down menu in the components control panel (right side of the screen).



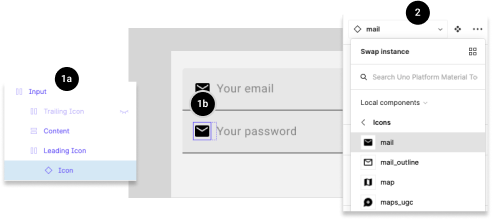
### What Are the Layers in a Text Box?

1. **Template:** Link to the main template
2. **Assistive text:** Text displayed under the text box
3. **Line:** Horizontal line under the text box (only in filled type text box)
4. **Input:** Group containing the Suffix, Icons, and Input text
5. **TrailingIcon:** Icon displayed on the right
6. **Content:** Group containing the Input text and Label
7. **Input text:** Group containing the Cursor, Text, and Prefix
8. **Cursor:** Vertical bar on the right of the text (used only for illustrative purposes)
9. **Text:** Text input by the user
10. **Prefix:** Text displayed on the left
11. **Label:** Name of the text box
12. **LeadingIcon:** Icon displayed on the left



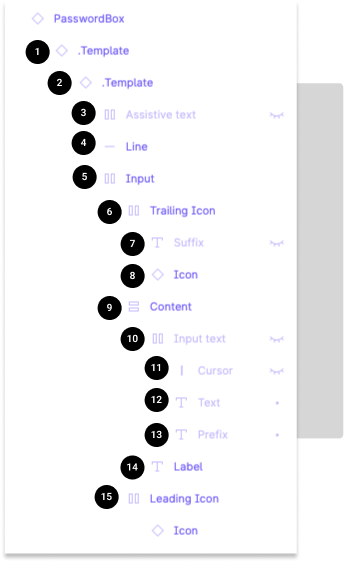
### How to Add a Password Box?

1. **Drag the Password Box** from the Text components library (left side of your screen) into your Figma project.
2. **Select the Password Box** by clicking on it. You should see a purple outline around the component.
3. **Customize the look and content** of the Password Box with the options you need from the variants control panel (right side of your screen).
4. **Also customize the look and content** of the Text Box template with the options you need from the variants control panel (right side of your screen). To have access to these options you must select the second . Template in the layer panel (left side of your screen)
5. **Adjust the width of the Password Box** by changing the value from "fixed width" to "fill container" in the Resizing panel (right side of your screen).
6. **Edit the text of the Password Box.** To do so, you could click on the label until the text becomes highlighted or select the label layer from the layers panel (left side of the screen). Once selected, use your keyboard to enter new text.
7. **Make the Assistive label invisible** if you are not using it.



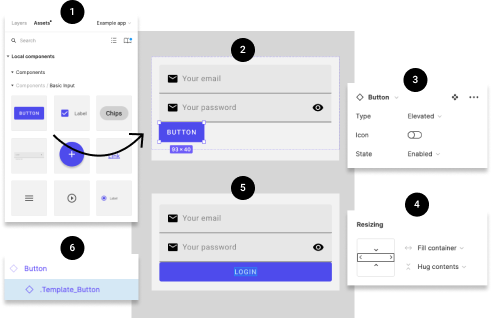
### How to Change the Icon of a Password Box?

1. **Select the icon** from a Password Box by clicking directly on the icon multiple times until the icon becomes highlighted (1b) or by selecting the Icon layer from the layer panel (left side of your screen - 1a).
2. **Choose a new icon** from the drop-down menu in the components control panel (right side of the screen).



### What Are the Layers in a Password Box?

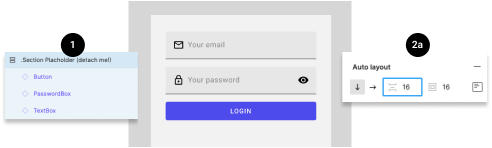
1. **Template 1:** Link to the main Password Box template
2. **Template 2:** Link to the Text Box template
3. **Assistive text:** Text displayed under the text box
4. **Line:** Horizontal line under the text box (only in filled type text box)
5. **Input:** Group containing the TraillingIcon, and content
6. **TrailingIcon:** Group containing the Suffix, and Icon
7. **Suffix:** Text displayed on the right
8. **Icon:** Icon displayed on the right
9. **Content:** Group containing the Input text and Label
10. **Input text:** Group containing the Cursor, Text, and Prefix
11. **Cursor:** Vertical bar on the right of the text (used only for illustrative purposes)
12. **Text:** Text input by the user
13. **Prefix:** Text displayed on the left
14. **Label:** Name of the text box
15. **LeadingIcon:** Icon displayed on the left



### How to Add a Button?

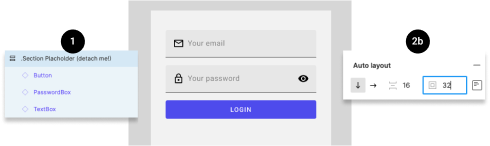
1. **Drag** the Button from the Basic Input components library (left side of your screen) into your Figma project.
2. **Select** the Button by clicking on it. You should see a purple outline around the component.
3. **Customize the look and content** of the Button with the options you need from the variants control panel (right side of your screen).
4. **Adjust the width** of the Button by changing the value from "fixed width" to "fill container" in the Resizing panel (right side of your screen). You will also need to repeat this action on the . Template\_Button (6).
5. **Edit the text** of the Button. To do so, you could click on the label until the text becomes highlighted or select the label layer from the layers panel (left side of the screen). Once selected, use your keyboard to enter new text.

## Step 3: Adjusting the layout



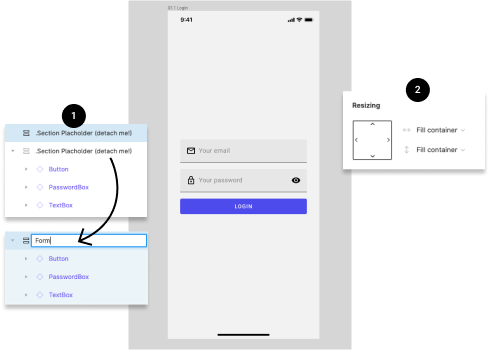
### How to Edit the Spacing Between the Elements?

1. **Select the layer group** from the layer panel on the left side of the screen.
2. **Go to the Auto layout panel** on the right side of the screen. Change the value of the spacing between elements.



### How to Edit the Padding Around the Items?

1. **Select the layer group** from the layer panel on the left side of the screen.
2. **Go to the Auto layout panel** on the right side of the screen. Change the value of the padding around the items.

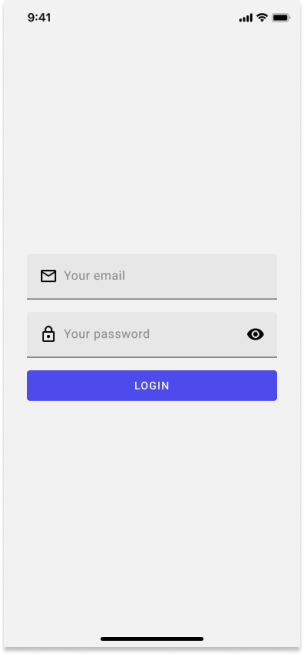


### How to Make the Content Centered Vertically?

1. For this example, I suggest you first delete the unused Section Placeholder from the layer panel (left side of your screen). Also, you should remain the layer containing the form.
2. Adjust the height of the form section by changing the value from "fixed height" to "fill container" in the Resizing panel (right side of your screen).

## You did it!

Your page is now complete, and should now look like this.



Reference:

<https://platform.uno/docs/articles/figma-plugin/designers/simple-login-page.html>

**TermWork 2 :**

**Problem definition: Create a Basic Web page and host it on GitHub and AWS**

Theory:

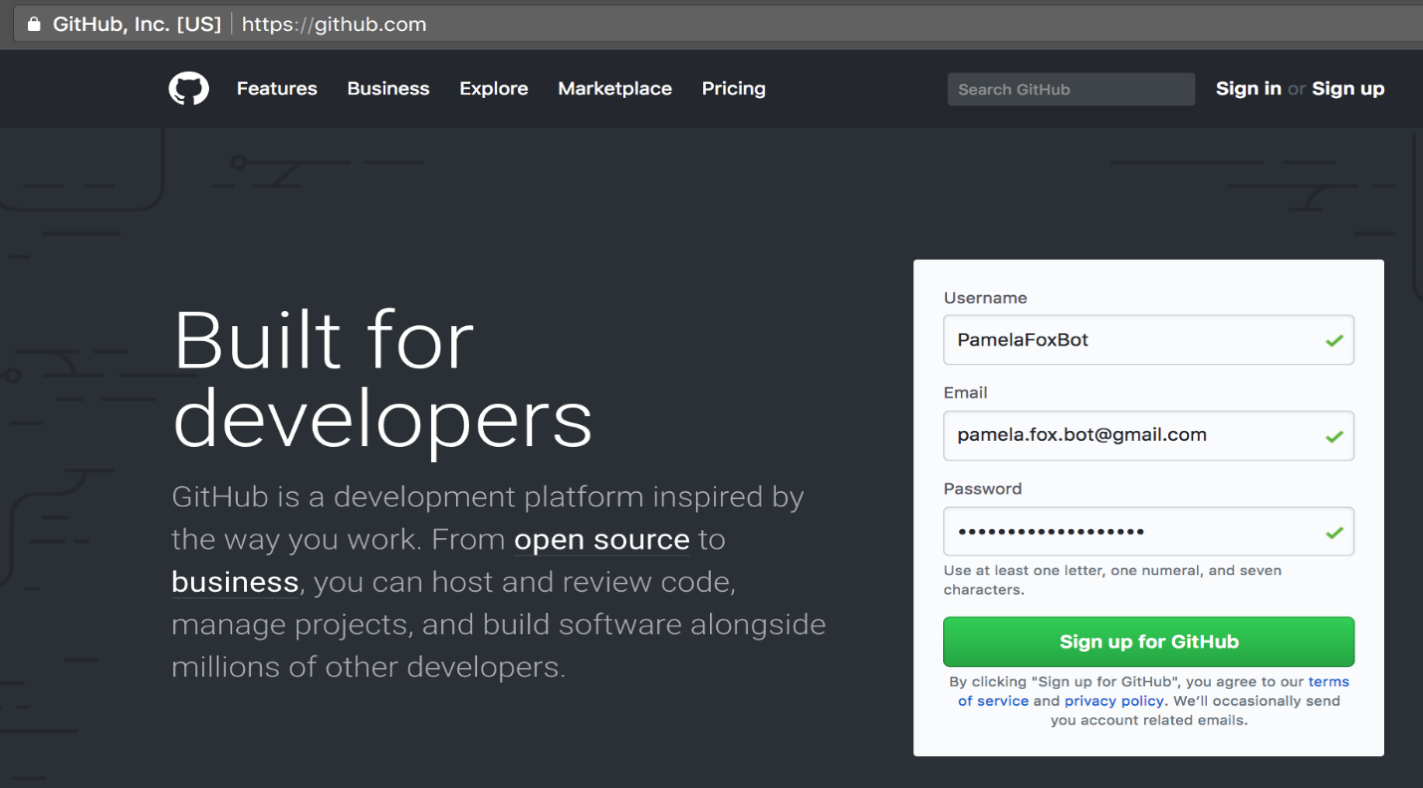
[Github](http://github.com/) is a company that hosts "code repositories", collections of code for projects. Many of the code repositories are "open source", which means they're publicly available for the whole world to browse. That's awesome, because that's how programmers learn from each other and build on each other's work! Khan Academy has quite a few [open source repositories](https://github.com/Khan). Code repositories can contain all sorts of types of code, not just HTML and CSS - whatever code is needed to make the project work.

Why am I telling you about them? Well, Github has a feature called [Github Pages](https://pages.github.com/), which makes it easy and free for you to create a multi-file website hosted at yourusername.github.io.

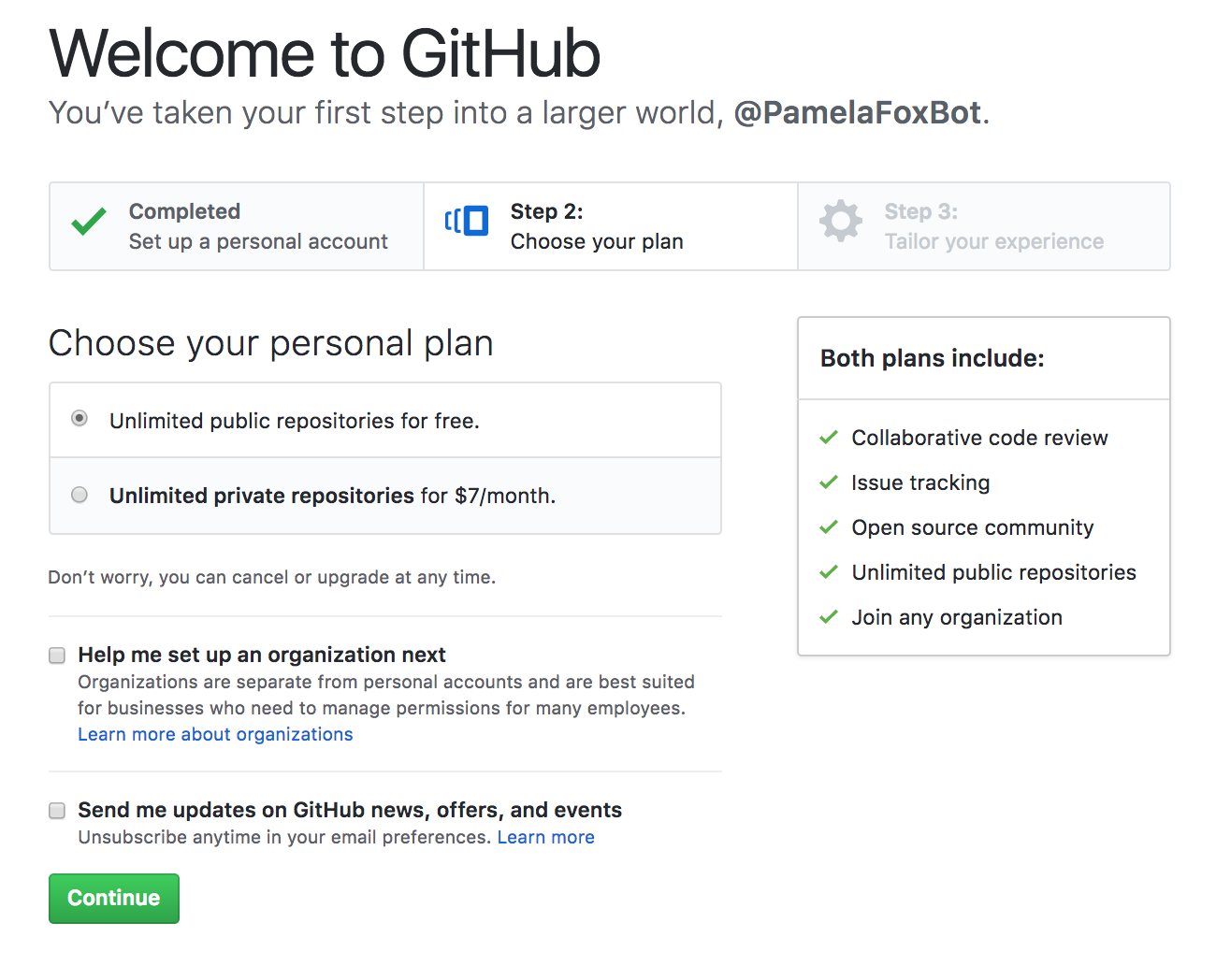
### **Sign up for a Github account**

Visit [github.com](http://github.com/), and you'll see a signup form on their front page. (If you don't, congrats, you already have an account! You can skip to Step 2).

When you're picking up your username, keep in mind that it will be publicly viewable on the internet and it will be in the URL of your website: yourusername.github.io. Some people like to use nicknames or funny names instead of their real name, for privacy reasons, but that's up to you.

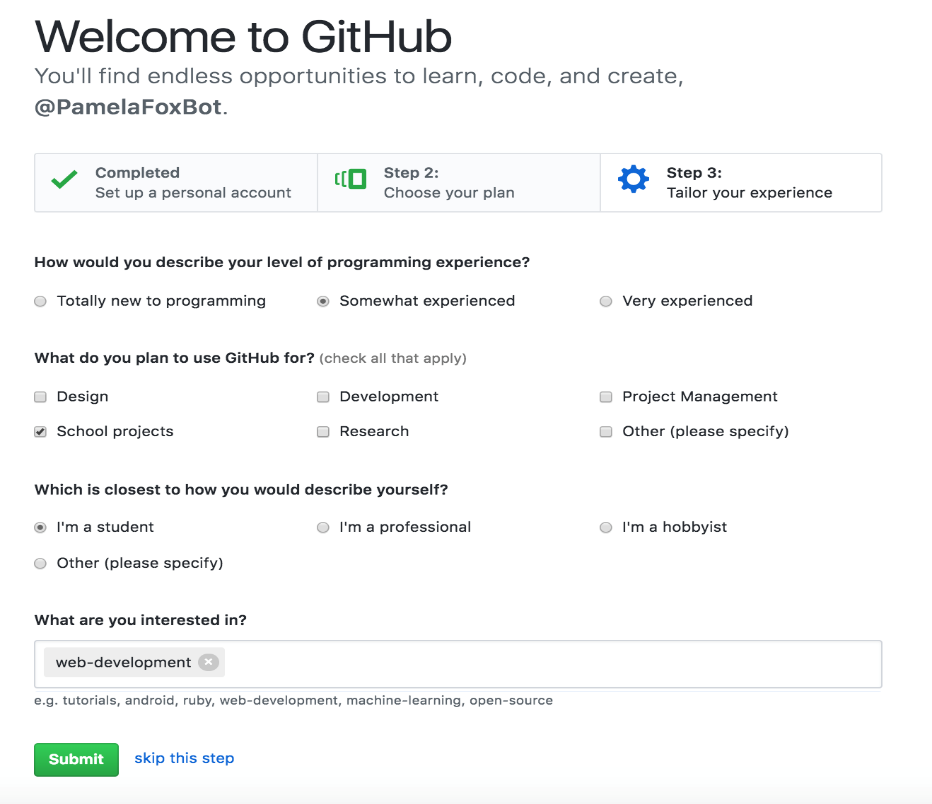
Screenshot of Github signup screen

On Step 2, **choose the free plan**. You don't need a paid plan for a public website repository. (You can always upgrade later if you become a prolific Github user and need their professional features.)



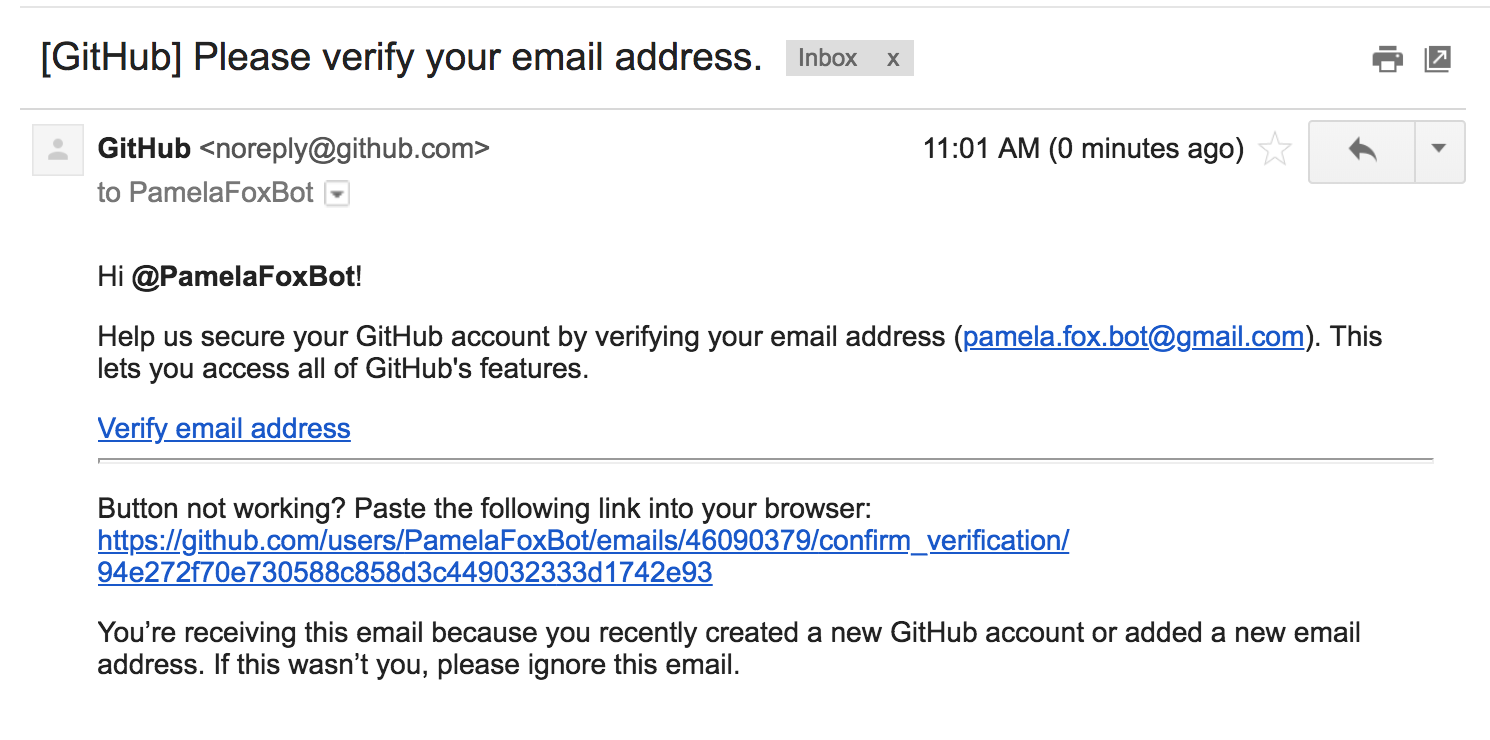
Screenshot of Github signup step 2

On Step 3, you can answer the survey questions or skip that step:



Screenshot of Github sign up process, step 3 survey

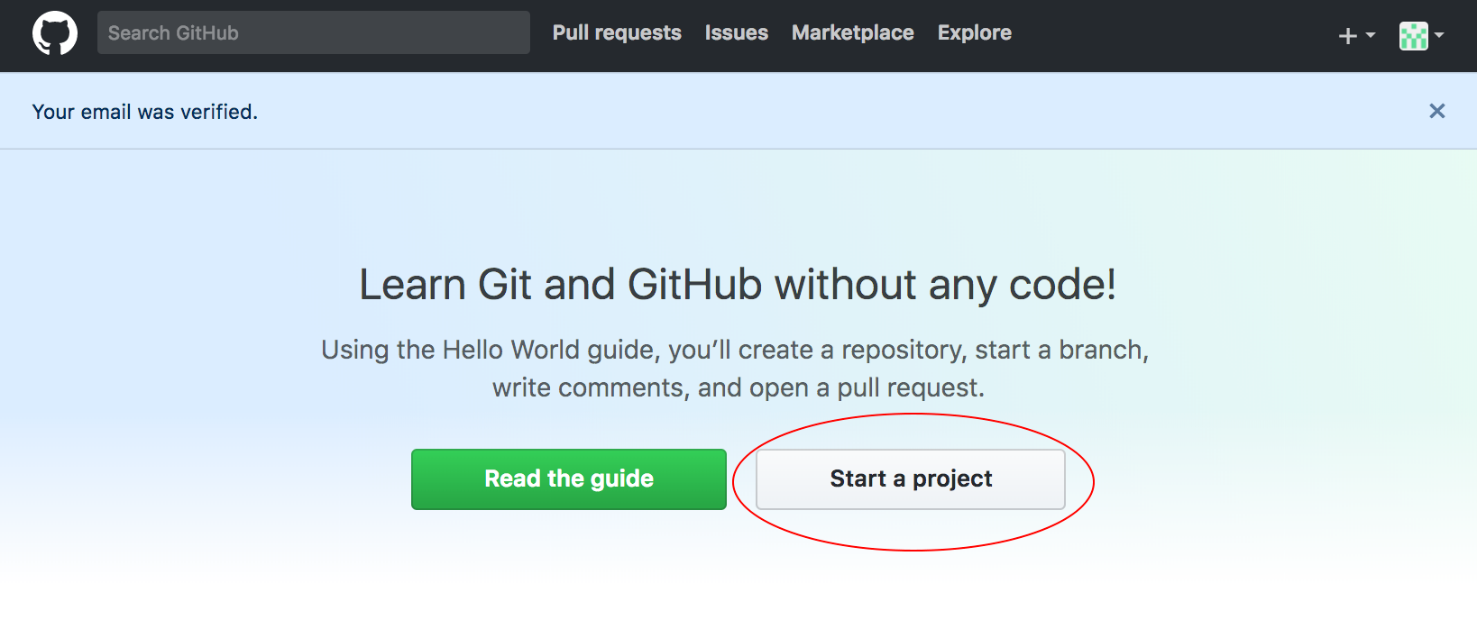
After you fill out the new account form, make sure you check your email inbox and **click the link** to verify your email address



Screenshot of Gmail verify screen for Github

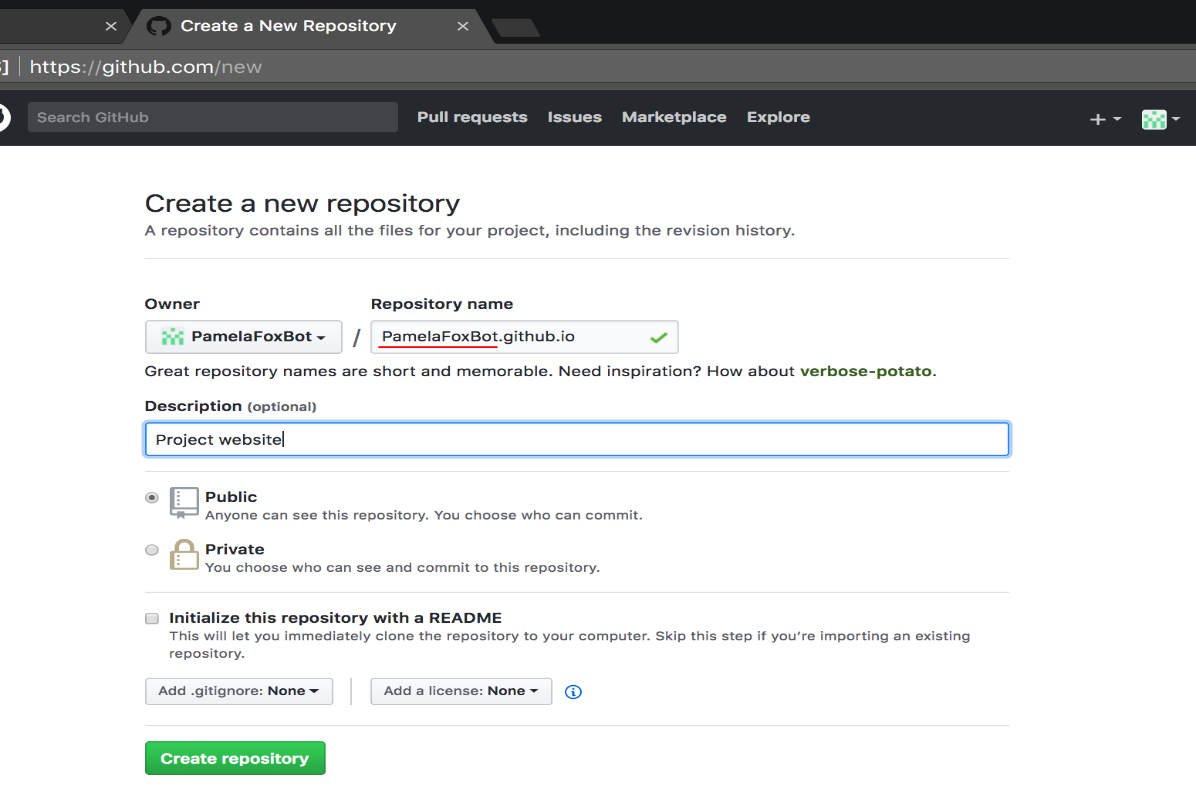
## Create a Github project for your website

Once you verify your email address, you'll see the start page. **Click "Start a project".**



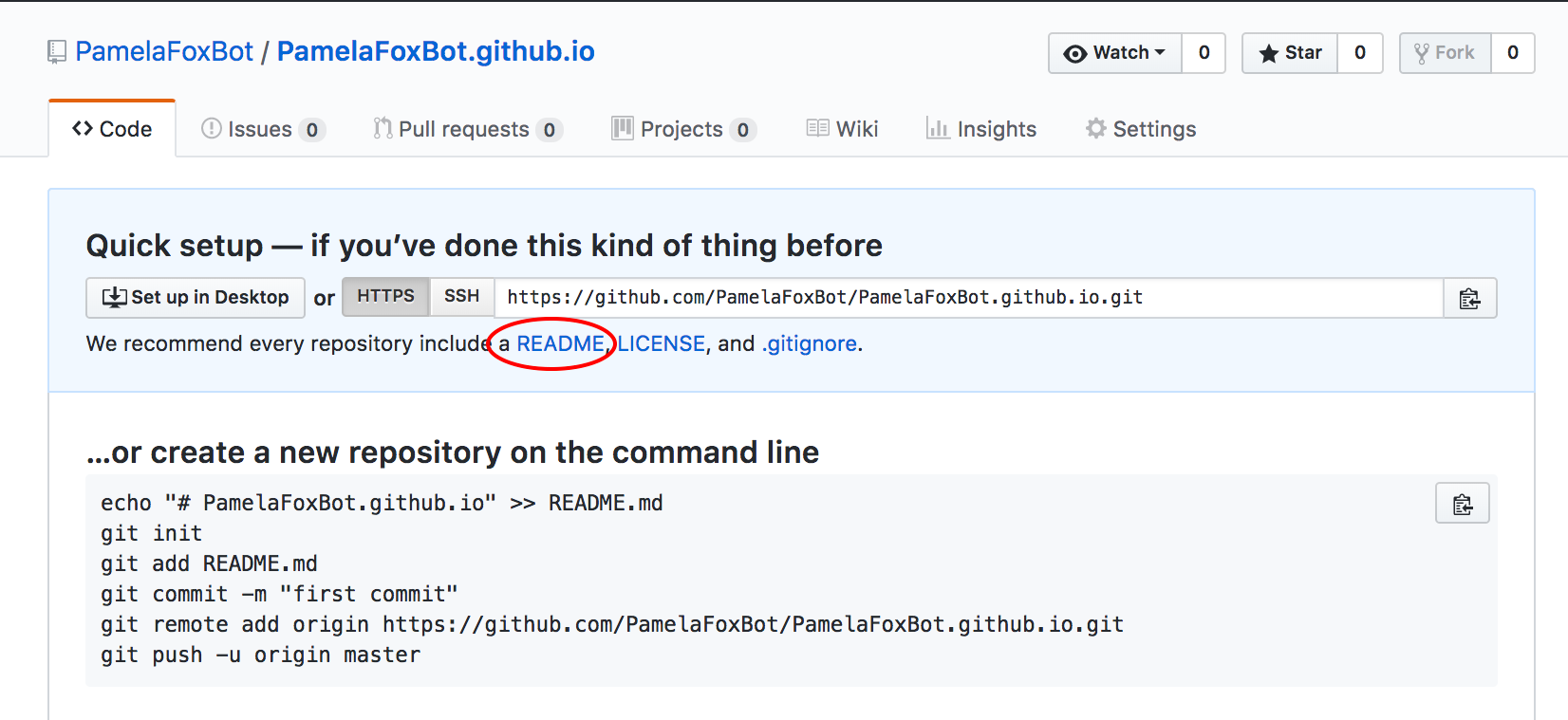
Screenshot of Github start page with two buttons, Read the guide and Start a project

On the project creation page, you'll be asked for a project name. **Make sure to name your project exactly “YOUR\_USERNAME.github.io”.** That tells Github that you’re making a special webpage project, so that it knows to upload your files to that user-facing URL whenever you change them. Since my username was "PamelaFoxBot", I named my project "PamelaFoxBot.github.io":



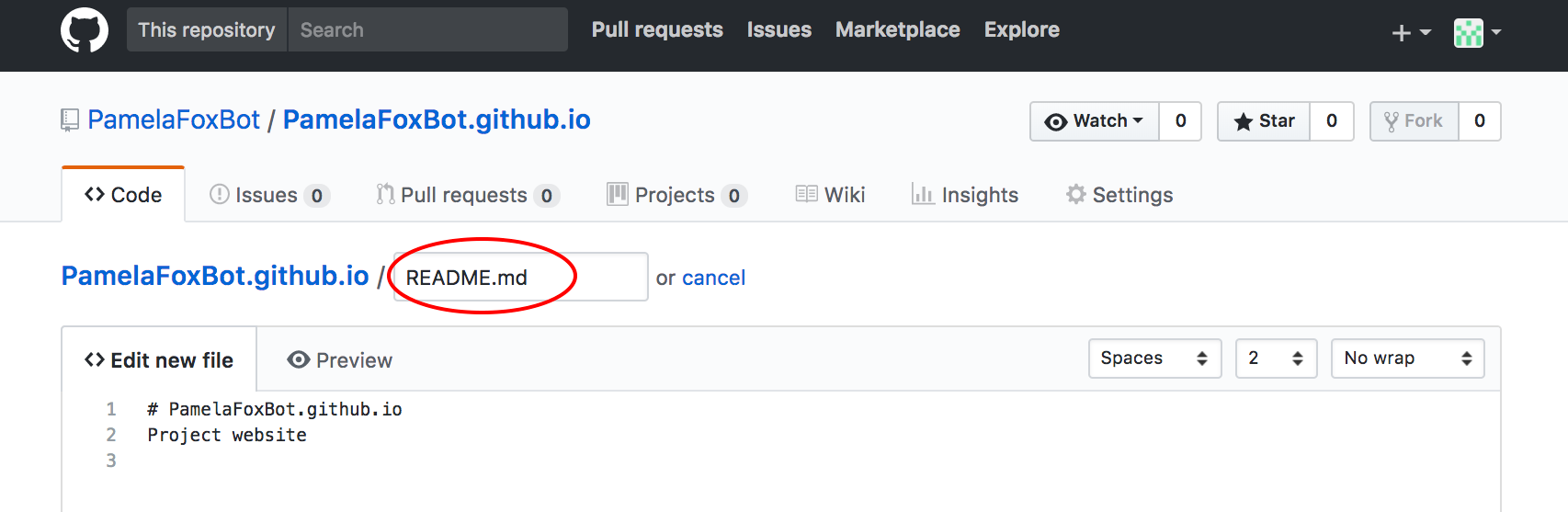
Screenshot of Github interface to create a new project

Congratulations, you have a new project! But it has no files yet. As a quick way to create the first file, **click the "README" link**:



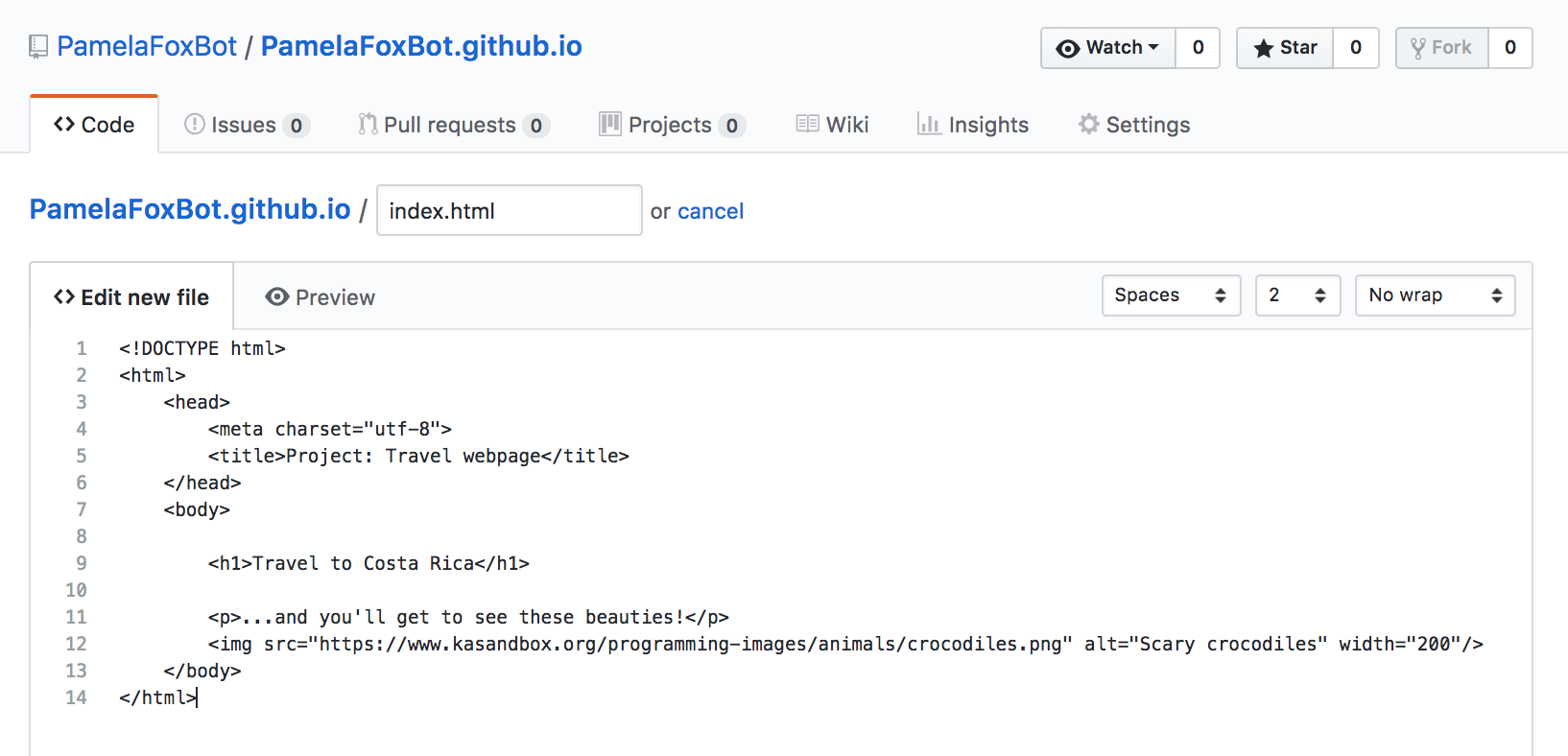
Screenshot of Github interface after making new project

Now you're looking at the Github file editor interface, for a file named README.md. **Change the filename from README.md to index.html.** Servers expect index.html to be the name of the main file of a website folder. Your index can then link to other pages of the website.



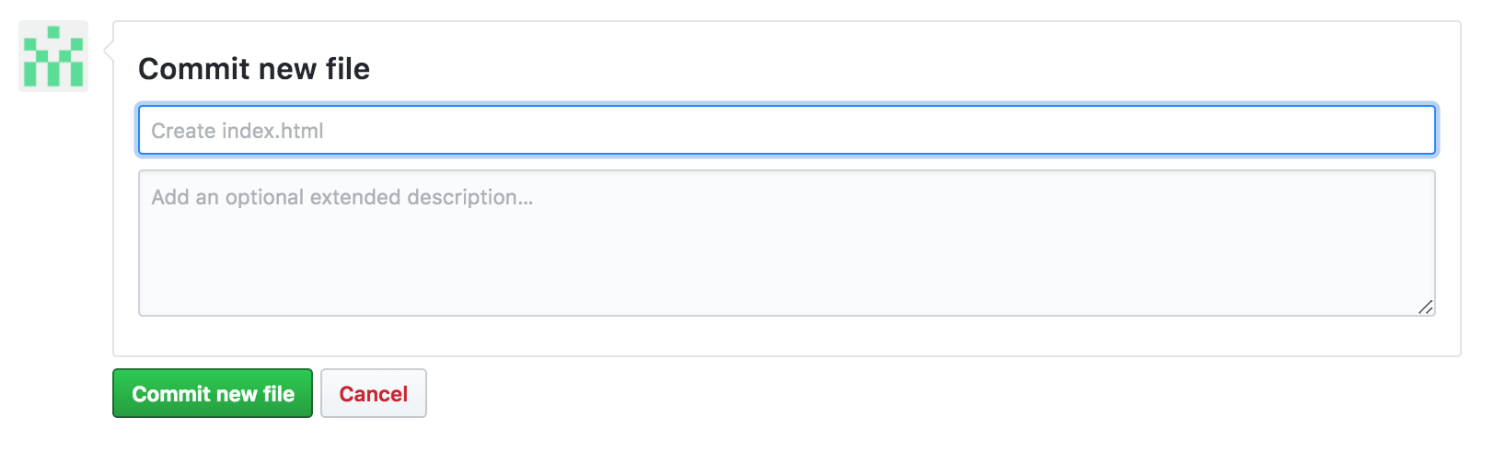
Screenshot of Github file editor for README.md, with filename circled

**Delete the current contents of the file editor and replace them with the HTML of your webpage.** You can copy and paste the HTML from a project from Khan Academy or one you've been developing on your computer.



Screenshot of Github file editor with HTML in the editing box

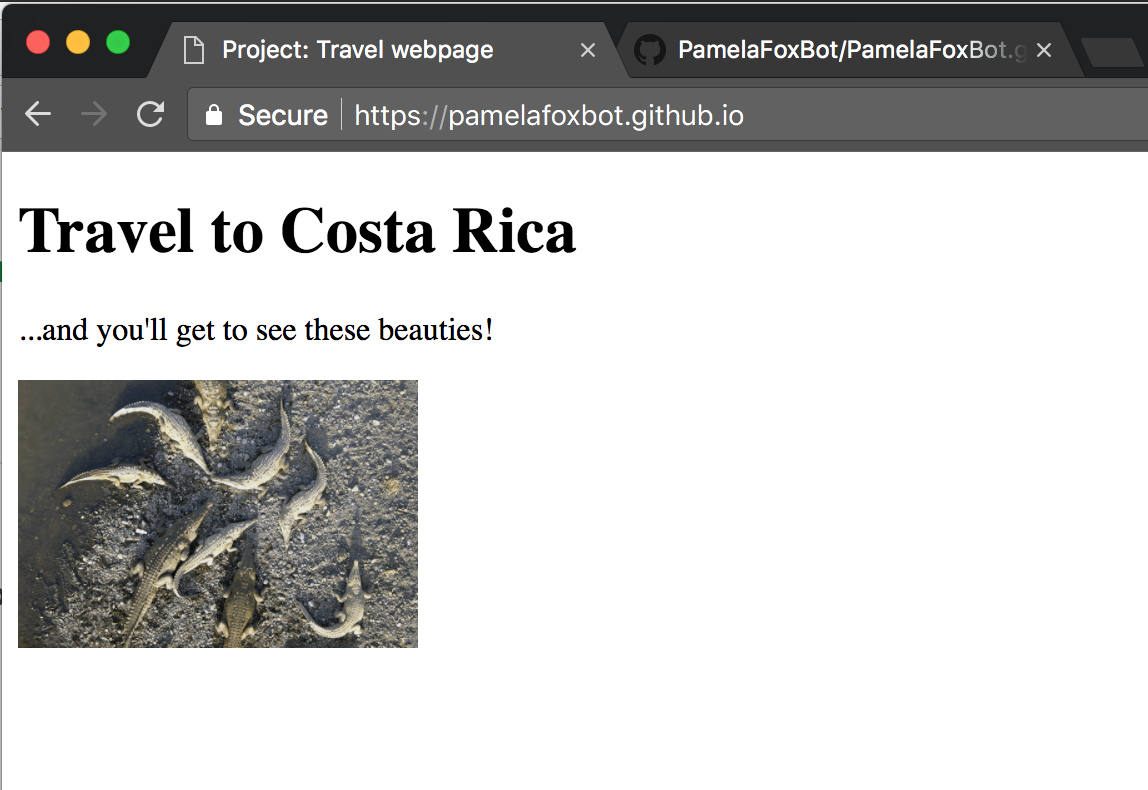
**Scroll down the page until you find the Commit area, and click “Commit new file”.** Every time you create a new version of a file, you are making a “commit” to record that in the file version history. Each commit has a message to describe the change. Github suggests a default message (“Create index.html”, in this case), but you can override that by typing in the box.



Screenshot of Github "Commit new file" interface

**Type YOUR\_USERNAME.github.io in the browser, and reload until your webpage appears.**

Github takes from 1-10 minutes to upload your project changes to the user-facing webpage URL, so you may need to exercise patience. You may also need to "hard reload" (hold shift while reloading), to clear the browser cache.



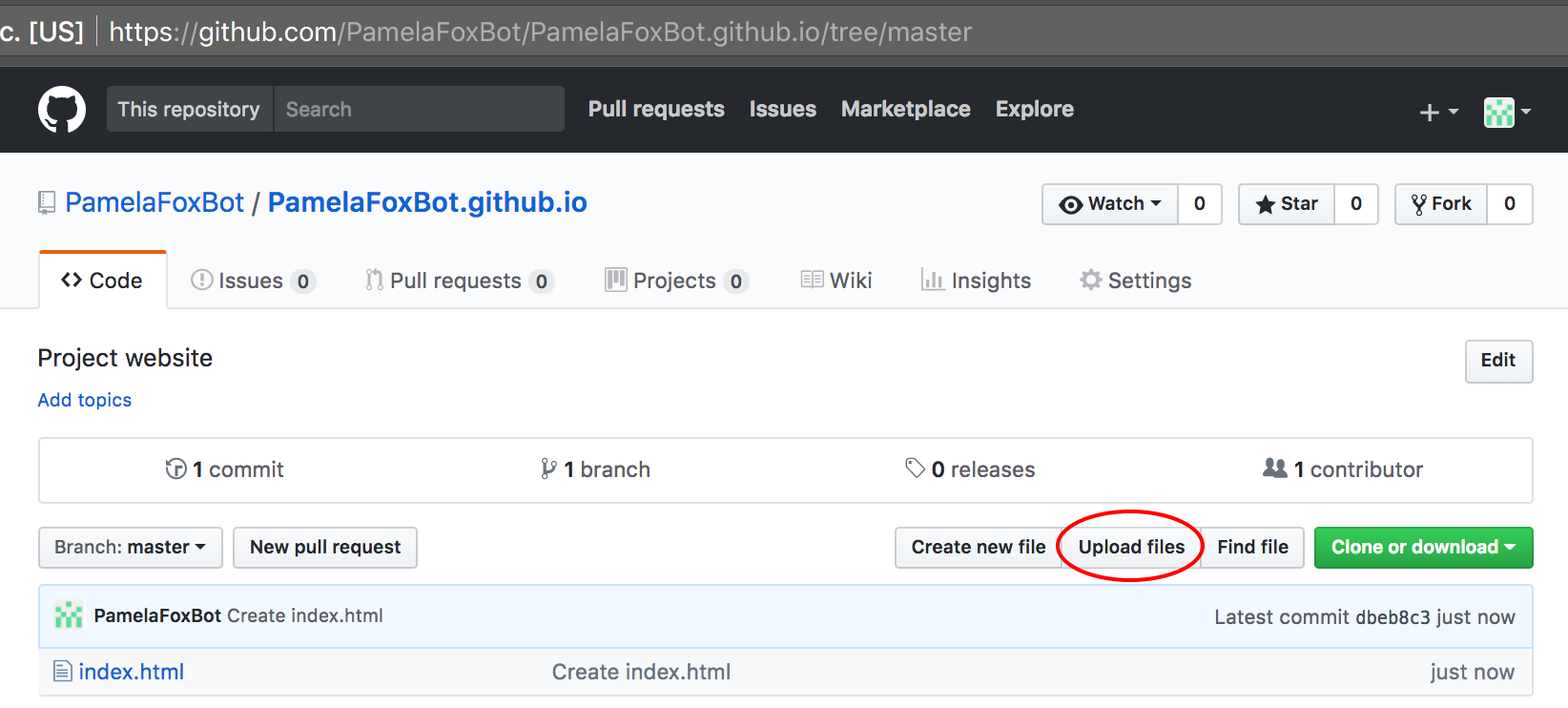
Screenshot of webpage on github.io

### Upload multi-file websites to Github

Some of you may already be working on multi-file websites outside of Khan Academy, using a desktop editor. Your website might have multiple HTML, CSS, JS, and image files.

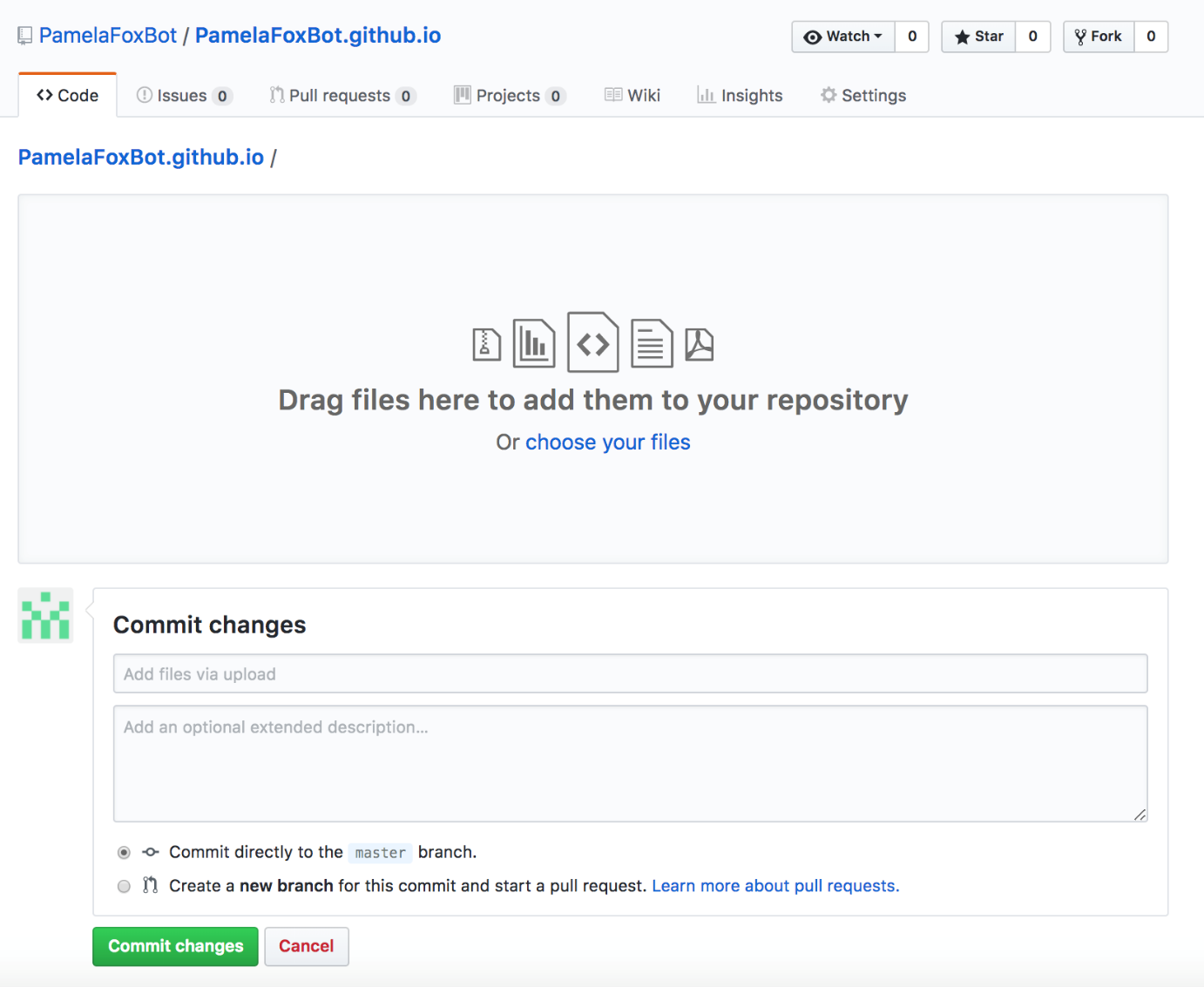
If you want to upload that website to Github, don't worry, you don't need to copy and paste each of those files individually! That would get quite tedious indeed. There's a better way...

**Open your project on Github and click "Upload files".**



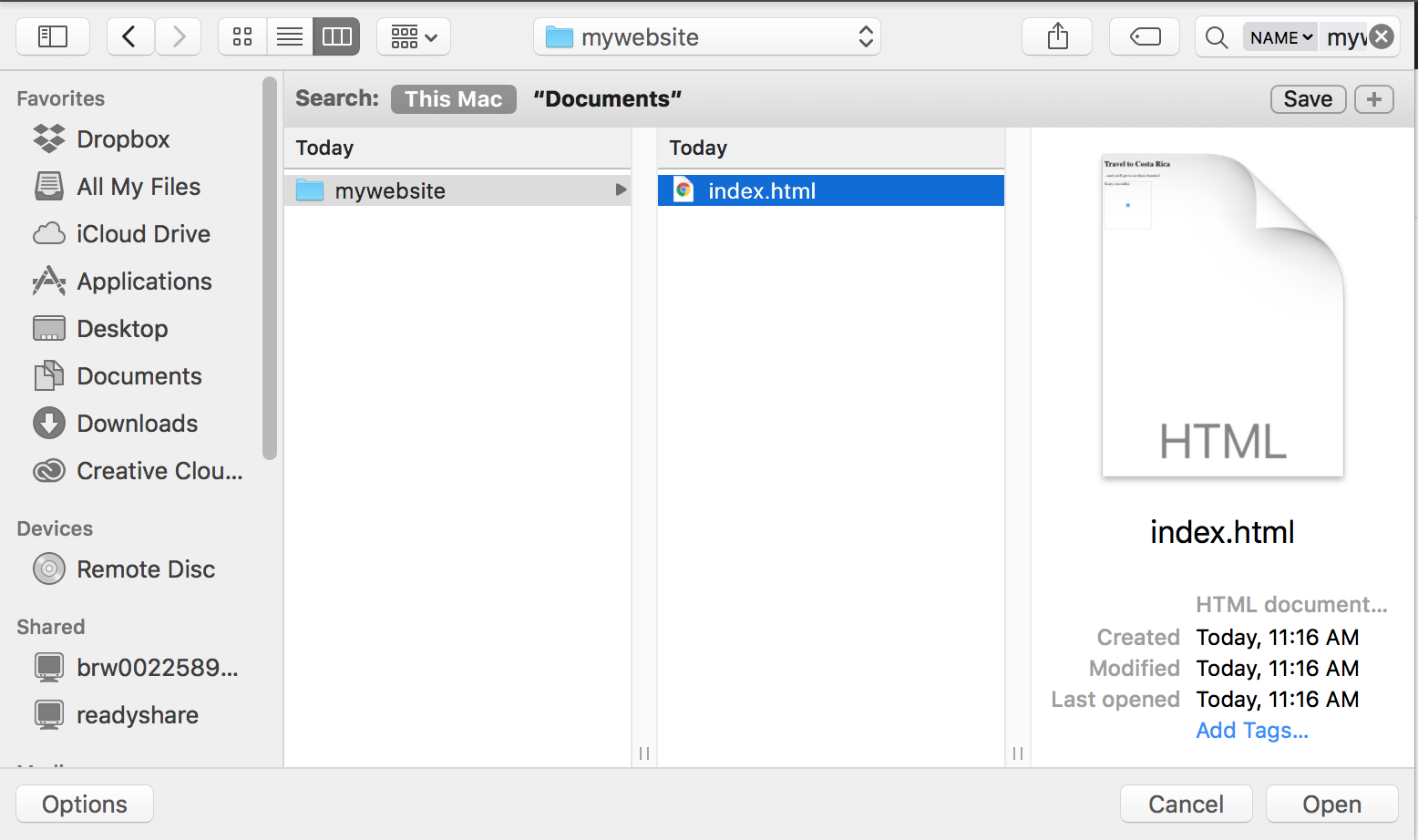
Screenshot of Github project interface with "Upload files" button circled red

You’ll see the Github file uploader interface. To select your files, **drag-and-drop or use the file chooser**:



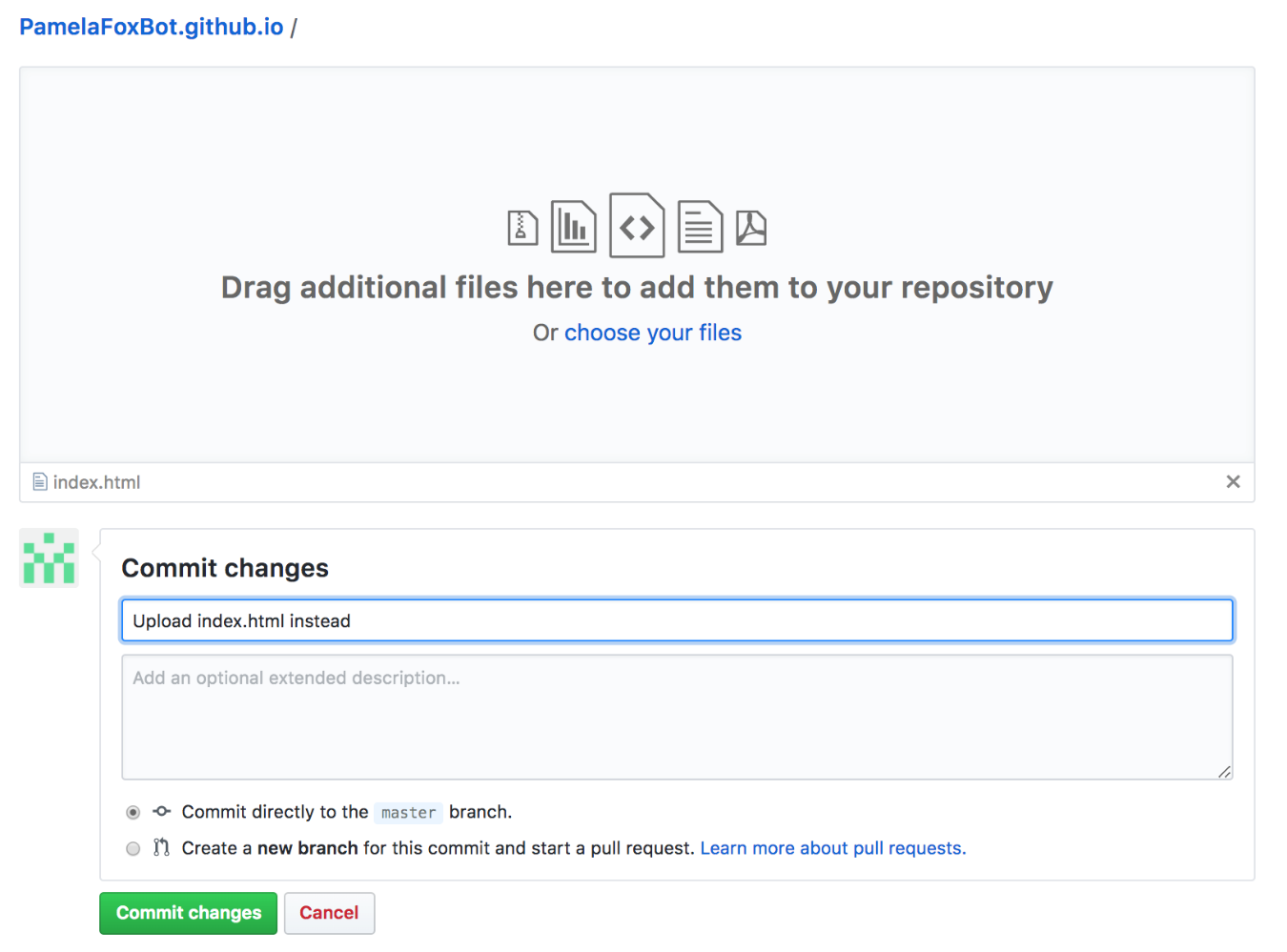
**Find your project folder and select the files you want to upload.**

Generally, you want to upload whatever files you’ve changed. If you’re not sure, just upload everything. Github will use an algorithm to figure out what has and hasn’t changed, and only make new versions for changed files.



Screenshot of file chooser

**Change the commit message as desired and click “Commit changes”.**

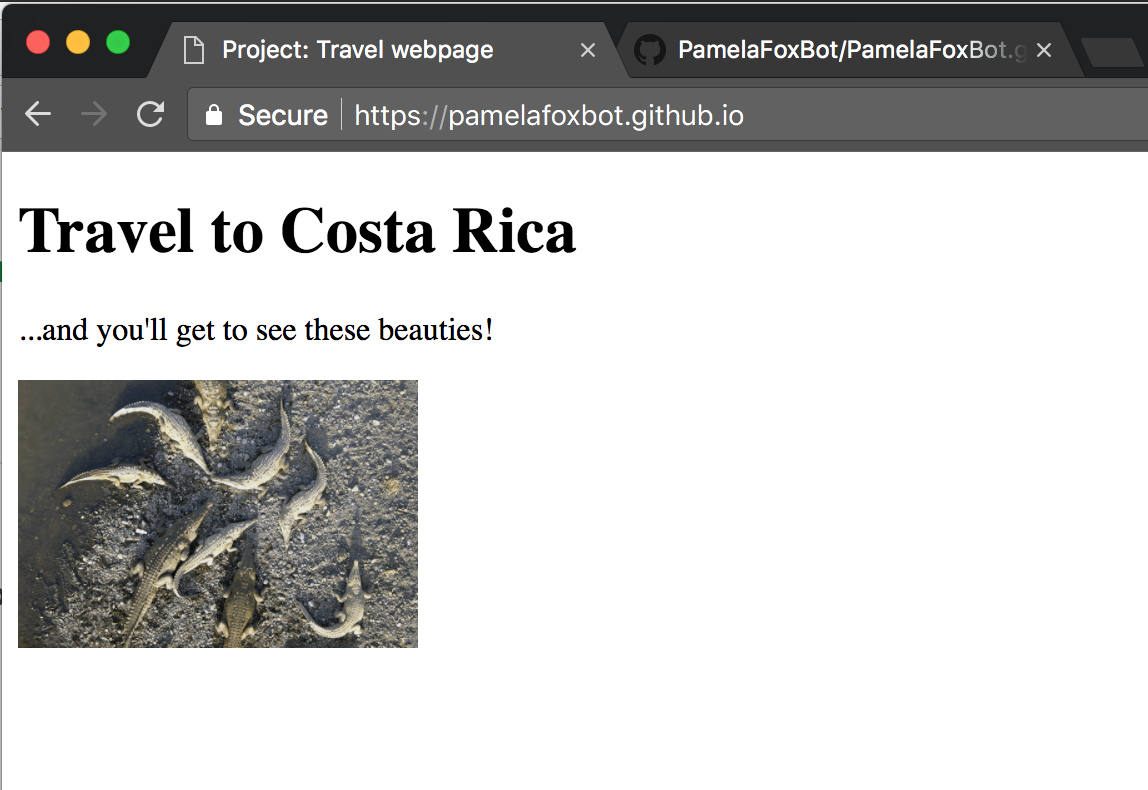


Screenshot of committing uploaded files

**Reload your webpage to see the changes!**

Remember that it takes up to 10 minutes. Patience is our friend here. 😊

That’s why it's best to do your development and testing in your local desktop environment, and only upload to Github (or any other hosting provider) once you’re happy with how your website looks locally.



Reference : <https://www.khanacademy.org/computing/computer-programming/html-css/web-development-tools/a/hosting-your-website-on-github>

<https://aws.amazon.com/getting-started/hands-on/host-static-website/>

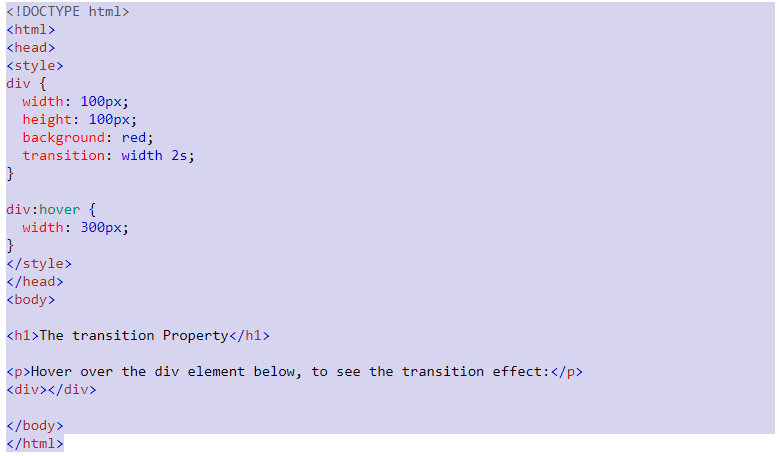
**Termwork 3:**

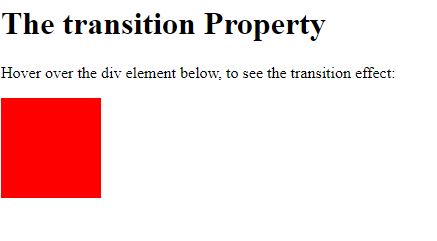
**Problem definition: Create animations and transition effects using CSS**

Theory:

The main goal of using the CSS is to style the web pages, make a good looking User interface. In this regard, we can add some animations and transitions effects which can beautify the appearance of the components of web pages.

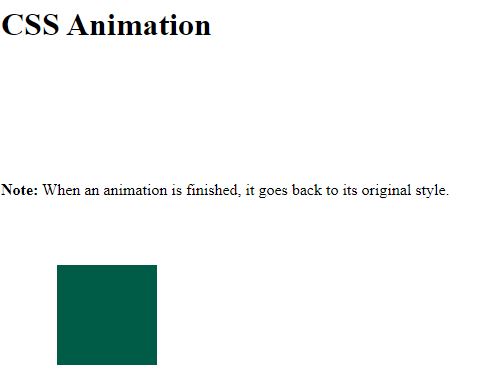
To create an animation using CSS





To create transitions effects using CSS





Reference :

1) <https://www.w3schools.com/css/tryit.asp?filename=trycss3_transition1>

2) <https://www.w3schools.com/css/tryit.asp?filename=trycss3_animation3>

**Termwork-04:**

**Problem definition: Create newspaper/magazine layout using CSS multi column Features.**

Theory:

CSS is the language we use to style an HTML document. CSS describes how HTML elements should be displayed. CSS stands for Cascading Style Sheets. CSS describes how HTML elements are to be displayed on screen, paper, or in other media. CSS saves a lot of work. It can control the layout of multiple web pages all at once. External stylesheets are stored in CSS files. A CSS rule consists of a selector and a declaration block. The selector points to the HTML element you want to style. The declaration block contains one or more declarations separated by semicolons. Each declaration includes a CSS property name and a value, separated by a colon. Multiple CSS declarations are separated with semicolons, and declaration blocks are surrounded by curly braces.



Source code:

<!DOCTYPE html>

<html>

<head>

<style>

.newspaper {

column-count: 3;

column-gap: 40px;

column-rule: 4px double #ff00ff;

}

</style>

</head>

<body>

<h1>The column-rule Property</h1>

<p>The column-rule property sets the width, style, and color of the rule between the columns of the element:</p>

<div class="newspaper">

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Nam liber tempor cum soluta nobis eleifend option congue nihil imperdiet doming id quod mazim placerat facer possim assum. Typi non habent claritatem insitam; est usus legentis in iis qui facit eorum claritatem. Investigationes demonstraverunt lectores legere me lius quod ii legunt saepius.

</div>

</body>

</html>

**Output:**

Reference:

https://www.w3schools.com/cssref/tryit.php?filename=trycss3\_column-rule

**Termwork-5:**

**Problem Definition: Create an application to demonstrate the use of Tailwind**

Theory:

Tailwind CSS is basically a Utility first CSS framework for building rapid custom UI. It is a highly customizable, low-level CSS framework that gives you all of the building blocks that you need. Also, it is a cool way to write inline styling and achieve an awesome interface without writing a single line of your own CSS. Tailwind has come with inbuilt a lot of features and styles for users to choose from and is also used to reduce the tendency of writing CSS code and create a beautiful custom UI. It will help you to overcome the complicated task. Tailwind CSS creates small utilities with a defined set of options enabling easy integration of existing classes directly into the HTML code.

Advantages of Tailwind CSS:

1. No more silly names for CSS classes and Id’s.
2. Minimum lines of Code in CSS file.
3. We can customize the designs to make the components.
4. Makes the website responsive.
5. Makes the changes in the desired manner.
6. CSS is global in nature and if make changes in the file the property is changed in all the HTML files linked to it. But with the help of Tailwind CSS we can use utility classes and make local changes.

Source Code:

<! Doctype html>

<head>

<title> Tailwind CSS Registration Form </title>

</head>

<html>

<body>

<h1 style= color:#abc123;> Tailwind CSS </h1>

<div class="block p-6 rounded-lg shadow-lg bg-white max-w-md">

<form>

<div class="grid grid-cols-2 gap-4">

<div class="form-group mb-6">

<input type="text" class="form-control

block

w-full

px-3

py-1.5

text-base

font-normal

text-gray-700

bg-white bg-clip-padding

border border-solid border-gray-300

rounded

transition

ease-in-out

m-0

focus:text-gray-700 focus:bg-white focus:border-blue-600 focus:outline-none" id="exampleInput123"

aria-describedby="emailHelp123" placeholder="First name">

</div>

<div class="form-group mb-6">

<input type="text" class="form-control

block

w-full

px-3

py-1.5

text-base

font-normal

text-gray-700

bg-white bg-clip-padding

border border-solid border-gray-300

rounded

transition

ease-in-out

m-0

focus:text-gray-700 focus:bg-white focus:border-blue-600 focus:outline-none" id="exampleInput124"

aria-describedby="emailHelp124" placeholder="Last name">

</div>

</div>

<div class="form-group mb-6">

<input type="email" class="form-control block

w-full

px-3

py-1.5

text-base

font-normal

text-gray-700

bg-white bg-clip-padding

border border-solid border-gray-300

rounded

transition

ease-in-out

m-0

focus:text-gray-700 focus:bg-white focus:border-blue-600 focus:outline-none" id="exampleInput125"

placeholder="Email address">

</div>

<div class="form-group mb-6">

<input type="password" class="form-control block

w-full

px-3

py-1.5

text-base

font-normal

text-gray-700

bg-white bg-clip-padding

border border-solid border-gray-300

rounded

transition

ease-in-out

m-0

focus:text-gray-700 focus:bg-white focus:border-blue-600 focus:outline-none" id="exampleInput126"

placeholder="Password">

</div>

<div class="form-group form-check text-center mb-6">

<input type="checkbox"

class="form-check-input appearance-none h-4 w-4 border border-gray-300 rounded-sm bg-white checked:bg-blue-600 checked:border-blue-600 focus:outline-none transition duration-200 mt-1 align-top bg-no-repeat bg-center bg-contain mr-2 cursor-pointer"

id="exampleCheck25" checked>

<label class="form-check-label inline-block text-gray-800" for="exampleCheck25">Subscribe to our newsletter</label>

</div>

<button type="submit" class="

w-full

px-6

py-2.5

bg-blue-600

text-white

font-medium

text-xs

leading-tight

uppercase

rounded

shadow-md

hover:bg-blue-700 hover:shadow-lg

focus:bg-blue-700 focus:shadow-lg focus:outline-none focus:ring-0

active:bg-blue-800 active:shadow-lg

transition

duration-150

ease-in-out">Sign up</button>

</form>

</div>

</body>

</html>

Output:

Reference:

https://tailwind-elements.com/docs/standard/forms/form-templates/?

**Termwork-6:**

**Problem definition: Create an application to demonstrate the use of JavaScript**

Theory:

JavaScript is the world's most popular programming language. JavaScript is the programming language of the Web. JavaScript is easy to learn. JavaScript is a lightweight, interpreted programming language. It is designed for creating network-centric applications. It is complimentary to and integrated with Java. JavaScript is very easy to implement because it is integrated with HTML. It is open and cross-platform.

Applications of Javascript Programming

As mentioned before, Javascript is one of the most widely used programming languages (Front-end as well as Back-end). It has it's presence in almost every area of software development. I'm going to list few of them here:

Client side validation - This is really important to verify any user input before submitting it to the server and Javascript plays an important role in validting those inputs at front-end itself.

Manipulating HTML Pages - Javascript helps in manipulating HTML page on the fly. This helps in adding and deleting any HTML tag very easily using javascript and modify your HTML to change its look and feel based on different devices and requirements.

User Notifications - You can use Javascript to raise dynamic pop-ups on the webpages to give different types of notifications to your website visitors.

Back-end Data Loading - Javascript provides Ajax library which helps in loading back-end data while you are doing some other processing. This really gives an amazing experience to your website visitors.

Presentations - JavaScript also provides the facility of creating presentations which gives website look and feel. JavaScript provides RevealJS and BespokeJS libraries to build a web-based slide presentation.

Server Applications - Node JS is built on Chrome's Javascript runtime for building fast and scalable network applications. This is an event based library which helps in developing very sophisticated server applications including Web Servers.

Source Code:

<!DOCTYPE html>

<html>

<head>

<script>

function validateForm() {

let x = document.forms["myForm"]["fname"].value;

if (x == "") {

alert("Name must be filled out");

return false;

}

}

</script>

</head>

<body>

<h2>JavaScript Validation</h2>

<form name="myForm" action="/action\_page.php" onsubmit="return validateForm()" method="post">

Name: <input type="text" name="fname">

<input type="submit" value="Submit">

</form>

</body>

</html>

Output:

Reference:

https://www.w3schools.com/js/tryit.asp?filename=tryjs\_validation\_j

**Termwork 7:**

**Problem definition: Create an application to demonstrate the use of Twitter Bootsrap**

**Theory:**

## *Basic Pagination*

If you have a web site with lots of pages, you may wish to add some sort of pagination to each page.

To create a basic pagination, add the .pagination class to an <ul> element. Then add the .page-item to each <li> element and a .page-link class to each link inside <li>.

## *Active State*

The .active class is used to "highlight" the current page:

## *Disabled State*

The .disabled class is used for un-clickable links:

## *Pagination Sizing*

Pagination blocks can also be sized to a larger or a smaller size:

Add class .pagination-lg for larger blocks or .pagination-sm for smaller blocks:

## *Pagination Alignment*

Use utility classes to change the alignment of the pagination:

<!-- Default (left-aligned) -->  
<ul class="pagination" style="margin:20px 0">  
  <li class="page-item">...</li>  
</ul>  
  
<!-- Center-aligned -->  
<ul class="pagination justify-content-center" style="margin:20px 0">  
  <li class="page-item">...</li>  
</ul>  
  
<!-- Right-aligned -->  
<ul class="pagination justify-content-end" style="margin:20px 0">  
  <li class="page-item">...</li>  
</ul>

## *Breadcrumbs*

Another form for pagination, is breadcrumbs:

[Photos](javascript:void(0)) / [Summer 2017](javascript:void(0)) / [Italy](javascript:void(0)) / Rome

The .breadcrumb and .breadcrumb-item classes indicates the current page's location within a navigational hierarchy:

<ul class="breadcrumb">  
  <li class="breadcrumb-item"><a href="#">Photos</a></li>  
  <li class="breadcrumb-item"><a href="#">Summer 2017</a></li>  
  <li class="breadcrumb-item"><a href="#">Italy</a></li>  
  <li class="breadcrumb-item active">Rome</li>  
</ul>

**Source Code:**

****

**Output:**

**References:**

[Bootstrap 4 Pagination (w3schools.com)](https://www.w3schools.com/bootstrap4/bootstrap_pagination.asp) <https://www.w3schools.com/bootstrap5/tryit.asp?filename=trybs_pagination&stacked=h>

**Termwork 8:**

**Problem Definition: Create an application to demonstrate the use of Jumbotron**

**Theory:**

A jumbotron indicates a big grey box for calling extra attention to some special content or information.

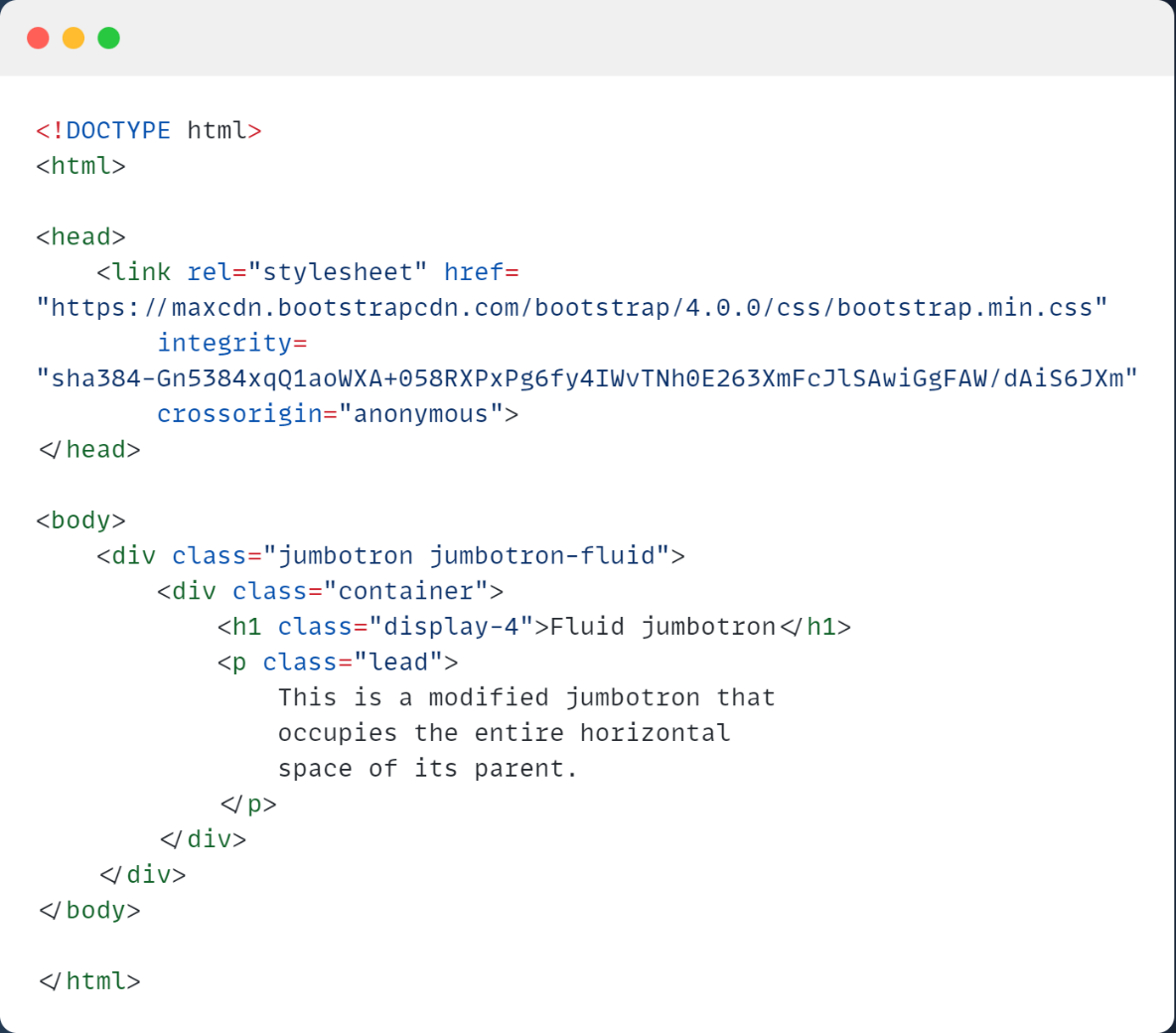
**Tip:** Inside a jumbotron you can put nearly any valid HTML, including other Bootstrap elements/classes.

Use a <div> element with class .jumbotron to create a jumbotron:

## Full-width Jumbotron

If you want a full-width jumbotron without rounded borders, add the .jumbotron-fluid class and a .container or .container-fluid inside of it:

**Source Code:**

****

**Output:**

**References:**

[Bootstrap 5 Jumbotron: 12 Examples (coder911.com)](https://www.coder911.com/bootstrap-5-jumbotron/) <https://www.coder911.com/bootstrap-5-jumbotron/>

**Termwork 9:**

**Problem Definition: Create an application to demonstrate the use of Web APIs**

**Theory:**

## *Locate the User's Position*

The HTML Geolocation API is used to get the geographical position of a user.

Since this can compromise privacy, the position is not available unless the user approves it.

**Note:**Geolocation is most accurate for devices with GPS, like smartphone.

The Geolocation API is supported in all browsers:

**Note:** As of Chrome 50, the Geolocation API will only work on secure contexts such as HTTPS. If your site is hosted on an non-secure origin (such as HTTP) the requests to get the users location will no longer function.

Geolocation is also very useful for location-specific information, like:

* Up-to-date local information
* Showing Points-of-interest near the user
* Turn-by-turn navigation (GPS)
* *Using the Geolocation API*

The getCurrentPosition() method is used to return the user's position.The Geolocation object also has other interesting methods:

* watchPosition() - Returns the current position of the user and continues to return updated position as the user moves (like the GPS in a car).
* clearWatch() - Stops the watchPosition() method.
* The getCurrentPosition() Method - Return Data
* The getCurrentPosition() method returns an object on success. The latitude, longitude and accuracy properties are always returned. The other properties are returned if available:

|  |  |
| --- | --- |
| Property | Returns |
| coords.latitude | The latitude as a decimal number (always returned) |
| coords.longitude | The longitude as a decimal number (always returned) |
| coords.accuracy | The accuracy of position (always returned) |
| coords.altitude | The altitude in meters above the mean sea level (returned if available) |
| coords.altitudeAccuracy | The altitude accuracy of position (returned if available) |
| coords.heading | The heading as degrees clockwise from North (returned if available) |
| coords.speed | The speed in meters per second (returned if available) |
| timestamp | The date/time of the response (returned if available) |

**Source Code:**

****

**Output:**

**References:**

[Geolocation API (w3schools.com)](https://www.w3schools.com/js/js_api_geolocation.asp) <https://www.w3schools.com/js/js_api_geolocation.asp>

**Termwork 10:**

**Problem Definition:** **Create an application to demonstrate the use of Ajax**

**Theory**

AJAX can be used for interactive communication with an XML file.

* *AJAX XML Example*

The following example will demonstrate how a web page can fetch information from an XML file with AJAX:

## *Example Explained*

When a user clicks on the "Get CD info" button above, the loadDoc() function is executed.

The loadDoc() function creates an XMLHttpRequest object, adds the function to be executed when the server response is ready, and sends the request off to the server.

When the server response is ready, an HTML table is built, nodes (elements) are extracted from the XML file, and it finally updates the element "demo" with the HTML table filled with XML data:

The XML file used in the example below looks like this: "[cd\_catalog.xml](https://www.w3schools.com/xml/cd_catalog.xml)".

<CATALOG>

<CD>

<TITLE>Empire Burlesque</TITLE>

<ARTIST>Bob Dylan</ARTIST>

<COUNTRY>USA</COUNTRY>

<COMPANY>Columbia</COMPANY>

<PRICE>10.90</PRICE>

<YEAR>1985</YEAR>

</CD>

<CD>

<TITLE>Hide your heart</TITLE>

<ARTIST>Bonnie Tyler</ARTIST>

<COUNTRY>UK</COUNTRY>

<COMPANY>CBS Records</COMPANY>

<PRICE>9.90</PRICE>

<YEAR>1988</YEAR>

</CD>

<CD>

<TITLE>Greatest Hits</TITLE>

<ARTIST>Dolly Parton</ARTIST>

<COUNTRY>USA</COUNTRY>

<COMPANY>RCA</COMPANY>

<PRICE>9.90</PRICE>

<YEAR>1982</YEAR>

</CD>

<CD>

<TITLE>Still got the blues</TITLE>

<ARTIST>Gary Moore</ARTIST>

<COUNTRY>UK</COUNTRY>

<COMPANY>Virgin records</COMPANY>

<PRICE>10.20</PRICE>

<YEAR>1990</YEAR>

</CD>

<CD>

<TITLE>Eros</TITLE>

<ARTIST>Eros Ramazzotti</ARTIST>

<COUNTRY>EU</COUNTRY>

<COMPANY>BMG</COMPANY>

<PRICE>9.90</PRICE>

<YEAR>1997</YEAR>

</CD>

<CD>

<TITLE>One night only</TITLE>

<ARTIST>Bee Gees</ARTIST>

<COUNTRY>UK</COUNTRY>

<COMPANY>Polydor</COMPANY>

<PRICE>10.90</PRICE>

<YEAR>1998</YEAR>

</CD>

<CD>

<TITLE>Sylvias Mother</TITLE>

<ARTIST>Dr.Hook</ARTIST>

<COUNTRY>UK</COUNTRY>

<COMPANY>CBS</COMPANY>

<PRICE>8.10</PRICE>

<YEAR>1973</YEAR>

</CD>

<CD>

<TITLE>Maggie May</TITLE>

<ARTIST>Rod Stewart</ARTIST>

<COUNTRY>UK</COUNTRY>

<COMPANY>Pickwick</COMPANY>

<PRICE>8.50</PRICE>

<YEAR>1990</YEAR>

</CD>

<CD>

<TITLE>Romanza</TITLE>

<ARTIST>Andrea Bocelli</ARTIST>

<COUNTRY>EU</COUNTRY>

<COMPANY>Polydor</COMPANY>

<PRICE>10.80</PRICE>

<YEAR>1996</YEAR>

</CD>

<CD>

<TITLE>When a man loves a woman</TITLE>

<ARTIST>Percy Sledge</ARTIST>

<COUNTRY>USA</COUNTRY>

<COMPANY>Atlantic</COMPANY>

<PRICE>8.70</PRICE>

<YEAR>1987</YEAR>

</CD>

<CD>

<TITLE>Black angel</TITLE>

<ARTIST>Savage Rose</ARTIST>

<COUNTRY>EU</COUNTRY>

<COMPANY>Mega</COMPANY>

<PRICE>10.90</PRICE>

<YEAR>1995</YEAR>

</CD>

<CD>

<TITLE>1999 Grammy Nominees</TITLE>

<ARTIST>Many</ARTIST>

<COUNTRY>USA</COUNTRY>

<COMPANY>Grammy</COMPANY>

<PRICE>10.20</PRICE>

<YEAR>1999</YEAR>

</CD>

<CD>

<TITLE>For the good times</TITLE>

<ARTIST>Kenny Rogers</ARTIST>

<COUNTRY>UK</COUNTRY>

<COMPANY>Mucik Master</COMPANY>

<PRICE>8.70</PRICE>

<YEAR>1995</YEAR>

</CD>

<CD>

<TITLE>Big Willie style</TITLE>

<ARTIST>Will Smith</ARTIST>

<COUNTRY>USA</COUNTRY>

<COMPANY>Columbia</COMPANY>

<PRICE>9.90</PRICE>

<YEAR>1997</YEAR>

</CD>

<CD>

<TITLE>Tupelo Honey</TITLE>

<ARTIST>Van Morrison</ARTIST>

<COUNTRY>UK</COUNTRY>

<COMPANY>Polydor</COMPANY>

<PRICE>8.20</PRICE>

<YEAR>1971</YEAR>

</CD>

<CD>

<TITLE>Soulsville</TITLE>

<ARTIST>Jorn Hoel</ARTIST>

<COUNTRY>Norway</COUNTRY>

<COMPANY>WEA</COMPANY>

<PRICE>7.90</PRICE>

<YEAR>1996</YEAR>

</CD>

<CD>

<TITLE>The very best of</TITLE>

<ARTIST>Cat Stevens</ARTIST>

<COUNTRY>UK</COUNTRY>

<COMPANY>Island</COMPANY>

<PRICE>8.90</PRICE>

<YEAR>1990</YEAR>

</CD>

<CD>

<TITLE>Stop</TITLE>

<ARTIST>Sam Brown</ARTIST>

<COUNTRY>UK</COUNTRY>

<COMPANY>A and M</COMPANY>

<PRICE>8.90</PRICE>

<YEAR>1988</YEAR>

</CD>

<CD>

<TITLE>Bridge of Spies</TITLE>

<ARTIST>T'Pau</ARTIST>

<COUNTRY>UK</COUNTRY>

<COMPANY>Siren</COMPANY>

<PRICE>7.90</PRICE>

<YEAR>1987</YEAR>

</CD>

<CD>

<TITLE>Private Dancer</TITLE>

<ARTIST>Tina Turner</ARTIST>

<COUNTRY>UK</COUNTRY>

<COMPANY>Capitol</COMPANY>

<PRICE>8.90</PRICE>

<YEAR>1983</YEAR>

</CD>

<CD>

<TITLE>Midt om natten</TITLE>

<ARTIST>Kim Larsen</ARTIST>

<COUNTRY>EU</COUNTRY>

<COMPANY>Medley</COMPANY>

<PRICE>7.80</PRICE>

<YEAR>1983</YEAR>

</CD>

<CD>

<TITLE>Pavarotti Gala Concert</TITLE>

<ARTIST>Luciano Pavarotti</ARTIST>

<COUNTRY>UK</COUNTRY>

<COMPANY>DECCA</COMPANY>

<PRICE>9.90</PRICE>

<YEAR>1991</YEAR>

</CD>

<CD>

<TITLE>The dock of the bay</TITLE>

<ARTIST>Otis Redding</ARTIST>

<COUNTRY>USA</COUNTRY>

<COMPANY>Stax Records</COMPANY>

<PRICE>7.90</PRICE>

<YEAR>1968</YEAR>

</CD>

<CD>

<TITLE>Picture book</TITLE>

<ARTIST>Simply Red</ARTIST>

<COUNTRY>EU</COUNTRY>

<COMPANY>Elektra</COMPANY>

<PRICE>7.20</PRICE>

<YEAR>1985</YEAR>

</CD>

<CD>

<TITLE>Red</TITLE>

<ARTIST>The Communards</ARTIST>

<COUNTRY>UK</COUNTRY>

<COMPANY>London</COMPANY>

<PRICE>7.80</PRICE>

<YEAR>1987</YEAR>

</CD>

<CD>

<TITLE>Unchain my heart</TITLE>

<ARTIST>Joe Cocker</ARTIST>

<COUNTRY>USA</COUNTRY>

<COMPANY>EMI</COMPANY>

<PRICE>8.20</PRICE>

<YEAR>1987</YEAR>

</CD>

</CATALOG>

**Source Code**

**:**

**Output:**

**References:**

[AJAX XML (w3schools.com)](https://www.w3schools.com/xml/ajax_xmlfile.asp) <https://www.w3schools.com/xml/ajax_xmlfile.asp>