Harvard University – Project0

<https://cs50.harvard.edu/web/2020/projects/0/search/>

[**Getting Started**](https://cs50.harvard.edu/web/2020/projects/0/search/#getting-started)

* Download the distribution code from <https://cdn.cs50.net/web/2020/spring/projects/0/search.zip> and unzip it. 

[**Specification**](https://cs50.harvard.edu/web/2020/projects/0/search/#specification)

Your website must meet the following requirements.

* **Pages**. Your website should have at least three pages: one for Google Search, one for Google Image Search, and one for Google Advanced Search. 
  + On the Google Search page, there should be links in the upper-right of the page to go to Image Search or Advanced Search. On each of the other two pages, there should be a link in the upper-right to go back to Google Search. ⮽
* **Query Text**. On the Google Search page, the user should be able to type in a query, click “Google Search”, and be taken to the Google search results for that page. ⮽
  + Like Google’s own, your search bar should be centered with rounded corners. The search button should also be centered, and should be beneath the search bar.
* **Query Images**. On the Google Image Search page, the user should be able to type in a query, click a search button, and be taken to the Google Image search results for that page. . 
* **Query Advanced**. On the Google Advanced Search page, the user should be able to provide input for the following four fields (taken from Google’s own [advanced search](https://www.google.com/advanced_search) options)
  + Find pages with… “all these words:” 
  + Find pages with… “this exact word or phrase:” ⮽
  + Find pages with… “any of these words:” 
  + Find pages with… “none of these words:” ⮽
* **Appearance**. Like Google’s own Advanced Search page, the four options should be stacked vertically, and all of the text fields should be left aligned.
  + Consistent with Google’s own CSS, the “Advanced Search” button should be blue with white text. When the “Advanced Search” button is clicked, the user should be taken to search results page for their given query. Must fix button
* **Lucky**. Add an “I’m Feeling Lucky” button to the main Google Search page. Consistent with Google’s own behavior, clicking this link should take users directly to the first Google search result for the query, bypassing the normal results page. 
* **Aesthetics**. The CSS you write should match Google’s own aesthetics as best as possible. . Unknown if it is doing it

[**Hints**](https://cs50.harvard.edu/web/2020/projects/0/search/#hints)

* To determine what the parameter names should be, you’re welcome to experiment with making Google searches, and looking at the resulting URL. It may also be helpful to open the “Network” inspector (accessible in Google Chrome by choosing View -> Developer -> Developer Tools) to view details about requests your browser makes to Google.
  + Any <input> element (whether its type is text, submit, number, or something else entirely) can have name and value attributes that will become GET parameters when a form is submitted. Unknown if it is doing it
  + You may also find it helpful to look at Google’s own HTML to answer these questions. In most browsers, you can control-click or right-click on a page and choose “View Page Source” to view the page’s underlying HTML.
* To include an input field in a form that users cannot see or modify, you can use a [“hidden”](https://www.w3schools.com/tags/att_input_type_hidden.asp) input field.

[**How to Submit**](https://cs50.harvard.edu/web/2020/projects/0/search/#how-to-submit)

**Hold on!** If you have already submitted and received a passing grade on the [**prior version of Project 0**](https://docs.cs50.net/web/2020/x/projects/0/project0.html) (Homepage), please stop here. You already have received an equivalence credit for this project, and you should not submit this assignment, as it will have no impact on your progress in the course and will therefore only slow our graders down!

1. Visit [this link](https://submit.cs50.io/invites/89679428401548238ceb022f141b9947), log in with your GitHub account, and click **Authorize cs50**. Then, check the box indicating that you’d like to grant course staff access to your submissions, and click **Join course**.
2. [Install Git](https://git-scm.com/downloads) and, optionally, [install submit50](https://cs50.readthedocs.io/submit50/).

When you submit your project, the contents of your web50/projects/2020/x/search branch should match the file structure of the unzipped distribution code as originally received. That is to say, your files should not be nested inside of any other directories of your own creation (search or project0, for example). Your branch should also not contain any code from any other projects, only this one. Failure to adhere to this file structure will likely result in your submission being rejected.

By way of example, for this project that means that if the grading staff visits https://github.com/me50/USERNAME/blob/web50/projects/2020/x/search/index.html (where USERNAME is your own GitHub username as provided in the form, below) your submission for index.html for this project should be what appears. If it doesn’t, reorganize your repository as needed to match this paradigm.

1. If you’ve installed submit50, execute
2. submit50 web50/projects/2020/x/search

Otherwise, using Git, push your work to https://github.com/me50/USERNAME.git, where USERNAME is your GitHub username, on a branch called web50/projects/2020/x/search.

1. [Record a screencast](https://www.howtogeek.com/205742/how-to-record-your-windows-mac-linux-android-or-ios-screen/) not to exceed 5 minutes in length, in which you demonstrate your project’s functionality. **Your URL bar must remain visible throughout your demonstration of the project.** Be certain that every element of the specification, above, is demonstrated in your video. There’s no need to show your code in this video, just your application in action; we’ll review your code on GitHub. [Upload that video to YouTube](https://www.youtube.com/upload) (as unlisted or public, but not private) or somewhere else. In your video’s description, you must timestamp where your video demonstrates each of the seven (7) elements of the specification. **This is not optional**, videos without timestamps in their description will be automatically rejected.
2. Submit [this form](https://forms.cs50.io/74d85c3d-c363-4344-994d-13816571eb2e).

You can then go to <https://cs50.me/cs50w> to view your current progress!