

We would like you to make a simple gallery application

- **Application Requirements:**
 - Show the photos in a grid view when the page loads
 - Support a fullscreen mode where if the user clicks on an image, it will show it in full fullscreen
 - Should work on commonly used devices: phone, tablet, desktop.
 - Only needs to support the chrome browser
 - Only plain JS is allowed, do not use any frameworks like React or JQuery.
 - You can use ES6 and polyfills if you really want to
 - It should query an api for a set of photos
 - Data is provided in photos.json but you must access it via through an http request as if it is an API endpoint. We recommend using the chrome web server (mentioned below) as the server to access this json data.
 - Please do NOT include tests in your submission, but please do write code that would be easy to test.
- **Also Required:** Along with your code, include one or two paragraphs and indicate what additional technical suggestions and changes you would include if you had more time and describe briefly how they could be implemented. I.e. if this was actually a project that will be released to all Redfin users in production. We will evaluate this along with your code.
 - Think about your technical setup, design, maintainability, and reusability. Don't just say add pagination.

This is estimated to take around 1.5 hours

Quickstart with chrome webserver::

You are free to use any web server you like. For simplicity we recommend using

<https://chrome.google.com/webstore/detail/web-server-for-chrome/ofhbbkphhbklhfoeikjpcbhemloocgigb?hl=en>

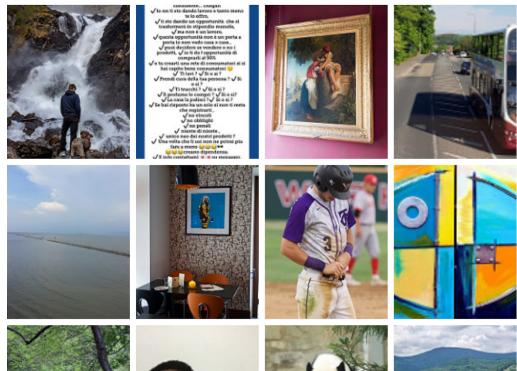
- 1) Download the [template code provided](#) and unzip into any folder ex: /code/myfolder
- 2) Launch the [chrome webserver](#)
- 3) Click choose folder and set to the /code/myfolder
- 4) Open a web page to <http://127.0.0.1:8887/index.html> and <http://127.0.0.1:8887/json-data/photos.json>, you should see a "Hello World" page and some json data. **Note that the json data could get old so some photos might no longer exist, we recommend you update by:**
 - a) go to <https://www.flickr.com/services/api/explore/flickr.photos.getRecent>

- b) in the "extras" argument field, add these: url_s, url_m, url_l
 - c) set the “per_page” argument to “50”
 - d) select json as the response and click "call method"
 - e) replace photo.json with the resulting data
- 5) Now build the gallery and it should load in index.html

See screen shots for some example mocks.

On page Load:

Gallary



Full Screen:

