

Automatic Slideshow With Transition (*nanogallery2*)

CISW400, Angelina Zwigl

What are we making with *nanogallery2*?

A photo gallery with captions that can be played as an automatic slideshow or manually clicked through! There is also a fade/sliding transition between each photo. →

MY CAT THROUGHOUT 2022!

Click on a photo and let the slideshow play (upper left corner) or click through manually!

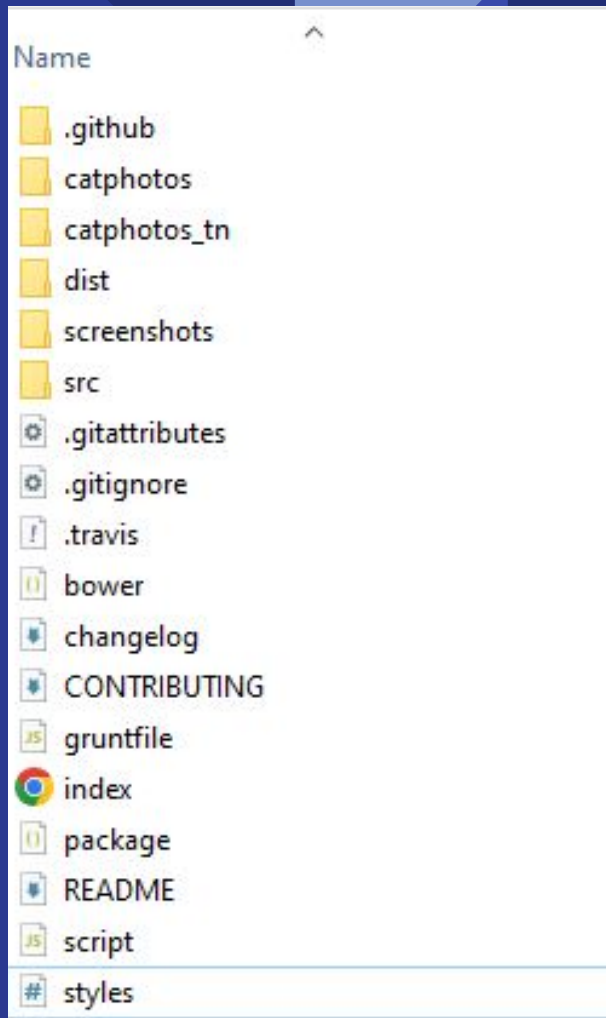


The Files

Here's all the files you downloaded!
Most of them will do all the nanogallery work for you.

We'll be working with the index, script, and styles.

Notice there's two folders of photos - the full resolutions (catphotos) and the thumbnails (catphotos_tn).



The Markup

```
<header>
  <h1>MY CAT THROUGHOUT 2022!</h1>
  <h2>Click on a photo and let the slideshow play
    (upper left corner) or click through manually!</h2>
</header>
<br>
  <div id="nanogallery2"></div>
```

The only necessary HTML markup for the gallery itself is a div with the id, #nanogallery. Don't put any content in the div, because it won't show up. Still add a main heading and subheading BEFORE the div to explain the content!

Styling the Header

Here are the styles for the header! Keep in mind that these styles are only good for a standard desktop screen.

Media queries would be necessary for other displays!

```
body { background-color: #DCDCD5;  
        font-family: "MS Sans Serif", Geneva, sans-serif; }  
header { background-color: #FEFFED;  
          background-repeat: no-repeat;  
          font-weight: bold;  
          height: 40%;  
          background-size: 100%;  
          color: #000000;  
          text-align: center;  
        }  
h1 { font-family: "sans-serif, Helvetica, Arial";  
      font-size: 5.0em;  
      padding-top: 20px; }  
h2 { font-size: 1.5em; }
```

jQuery

```
<meta name="viewport" content="user-scalable=no, width=device-width, initial-scale=1, maximum-scale=1">
<!-- stylesheet -->
<link href="styles.css" rel="stylesheet">
<!-- jQuery -->
<script src="https://cdn.jsdelivr.net/npm/jquery@3.3.1/dist/jquery.min.js" type="text/javascript"></script>
<!-- nanogallery2 -->
```

Add jQuery after your CSS stylesheet (if you have one).

Copy and paste from here and put in script tags:

<https://cdn.jsdelivr.net/npm/jquery@3.3.1/dist/jquery.min.js>

nanogallery2!

```
<!-- jQuery -->
<script src="https://cdn.jsdelivr.net/npm/jquery@3.3.1/dist/jquery.min.js" type="text/javascript"></script>
<!-- nanogallery2 -->
<link href="https://cdn.jsdelivr.net/npm/nanogallery2@3/dist/css/nanogallery2.min.css" rel="stylesheet" type="text/css">
<script type="text/javascript" src="https://cdn.jsdelivr.net/npm/nanogallery2@3/dist/jquery.nanogallery2.min.js"></script>
</head>
```

After the jQuery, add the nanogallery2 CSS and jQuery!

Copy and paste from here and put in script tags:

<https://cdn.jsdelivr.net/npm/nanogallery2@3/dist/css/nanogallery2.min.css>

<https://cdn.jsdelivr.net/npm/nanogallery2@3/dist/jquery.nanogallery2.min.js>

script.JS

Open your script.js file! This is a bit unusual, but we will put the images for the slideshow here as a variable:

```
var pictvar = [  
  { src: 'photo1.jpg', srct: 'photo1t.jpg', title: 'January' },  
  { src: 'photo2.jpg', srct: 'photo2t.jpg', title: 'February' },  
  { src: 'photo3.jpg', srct: 'photo3t.jpg', title: 'March' },  
  { src: 'photo4.jpg', srct: 'photo4t.jpg', title: 'April' },  
  { src: 'photo5.jpg', srct: 'photo5t.jpg', title: 'May' },  
  { src: 'photo6.jpg', srct: 'photo6t.jpg', title: 'June' },  
  { src: 'photo7.jpg', srct: 'photo7t.jpg', title: 'July' },  
  { src: 'photo8.jpg', srct: 'photo8t.jpg', title: 'August' },  
  { src: 'photo9.jpg', srct: 'photo9t.jpg', title: 'September' },  
  { src: 'photo10.jpg', srct: 'photo10t.jpg', title: 'October' }  
]
```

“src” is the full resolution photo and “srct” is its thumbnail! We can customize the height/width with the thumbnails and the page loads much quicker.

nanogallery2 Script

- Put your code inside a `$(window).on("load", function() { ... })` so your JavaScript will only run once the entire page has run.
- Inside the window load function, code the selector and another function for nanogallery2: `$("#nanogallery2").nanogallery2({ ... })`.

The rest of our nanogallery script will be inside BOTH of these functions!

```
$( window ).on( "load", function(){  
    $("#nanogallery2").nanogallery2( {
```

Gallery Settings & Content

Here, we can specify the dimensions we want displayed on the grid.

(This won't affect the dimensions on the actual slideshow!)

Try out a few heights and widths until you get something you like! This is nice, because it doesn't distort the photos like HTML and just crops them.

Then, copy and paste the photos' file location for the *itemsBaseURL*.

Should be very similar to what's in the search bar for index.html!

```
$( window ).on( "load", function(){
  $("#nanogallery2").nanogallery2( {
    // ### gallery settings ###
    thumbnailHeight: 300,
    thumbnailWidth: 300,
    itemsBaseURL: 'C:/Users/Angelina/Downloads/cisw400/zwiglmidterm/catphotos/',
    // ### gallery content ###
    items: pictvar
  });
});
```

Lastly, attach pictvar as the items and close both functions

Check the Console

You should have your gallery and slideshow up and working!

Click on the photos, play the slideshow, zoom in and out...

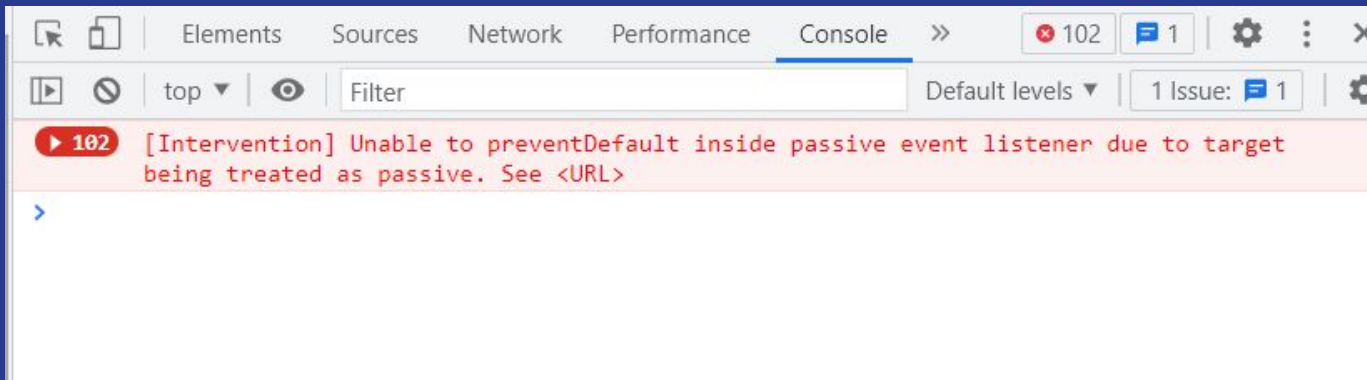
Check the Console for any issues - there is definitely a problem!

This is an issue with preventDefault in the js/min files.

Here are 2 possible fixes that might fix it, but I haven't had any luck! →

```
$( window ).on( "load", function(){  
  $("#nanogallery2").nanogallery2( {  
    // ### gallery settings ###  
    thumbnailHeight: 300,  
    thumbnailWidth: 300,  
    itemsBaseURL: 'C:/Users/Angel',  
    // ### gallery content ###  
    items: pictvar  
  }, {passive: false});  
});
```

```
#nanogallery2 {  
  touch-action: none; }
```

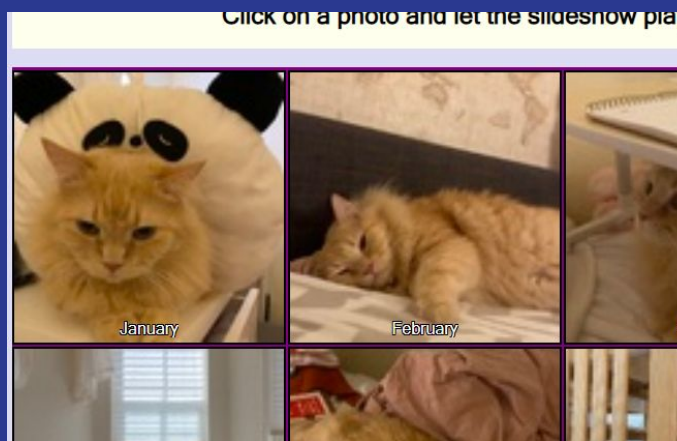


Playing With the Gallery

nanogallery has so many features you can play with!

I created some buttons to mess with the borders around the images -

```
<body>
  <header>
    <h1>MY CAT THROUGHOUT 2022!</h1>
    <button id="cream">cream border</button>
    <button id="darkblue">dark blue border</button>
    <button id="purple">purple border</button>
    <button id="black">black border</button>
    <h2>Click on a photo and let the slideshow play (u
  </header>
  <br>
```



Now the JavaScript...

Create a variable for each button and an addEventListener for when you click the button.

This looks like a lot of extra work, but it's really a lot of copying and pasting!

```
var btn1 = document.querySelector('#cream');
var btn2 = document.querySelector('#darkblue');
var btn3 = document.querySelector('#purple');
var btn4 = document.querySelector('#black');
btn1.addEventListener('click', function(){
  document.querySelector('div').style.backgroundColor = "#F5F5DC";
});
btn2.addEventListener('click', function(){
  document.querySelector('div').style.backgroundColor = "#32527B";
});
btn3.addEventListener('click', function(){
  document.querySelector('div').style.backgroundColor = "purple";
});
btn4.addEventListener('click', function(){
  document.querySelector('div').style.backgroundColor = "black";
});
```

Styling the Buttons

...and their styles!

```
button { height: 10%;
         font-size: 120%;
         border: none;
         background-color: transparent;
         font-family: inherit;
         padding: 0;
         cursor: pointer; }
#cream { float: left;
         margin-left: 100px;
         color: ■ #32527B; }
#darkblue { margin-right: 100px;
            padding-right: 400px;
            color: ■ #32527B; }
#purple { margin-right: px;
           color: ■ purple; }
#black { float: right;
         margin-right: 100px;
         color: ■ purple; }
```

Done!

