11.4.1

Gaining a Deeper Understanding of a Website Structure

Now that she knows basic HTML and CSS, Robin wants to find specific data. But on a webpage with lots of content, finding the HTML element that contains the data she wants can be overwhelming. So, she'll use Chrome Developer Tools (DevTools). With DevTools, developers can review the structure of any webpage. And not only that, but it also has a search function. This will help make sense of the tags and elements holding the data that Robin is seeking.

Specifically, Robin needs to identify elements of a webpage so that she can extract data from the News – NASA Mars Exploration (https://mars.nasa.gov/news/?page=0&per_page=40&order=publish_date+desc%2Ccreated_at+desc&search=&category=19%2C165%2C184%2C204&blank_scope=Latest) website. Let's follow along with her.

You've already learned how to identify and create basic HTML code in preparation for web scraping. You're now ready to learn more advanced skills to further your knowledge of how a website is structured. You'll follow the same process for each webpage that you want to scrape: visit the page, identify the data, then go through the HTML code to pinpoint its location on the webpage.

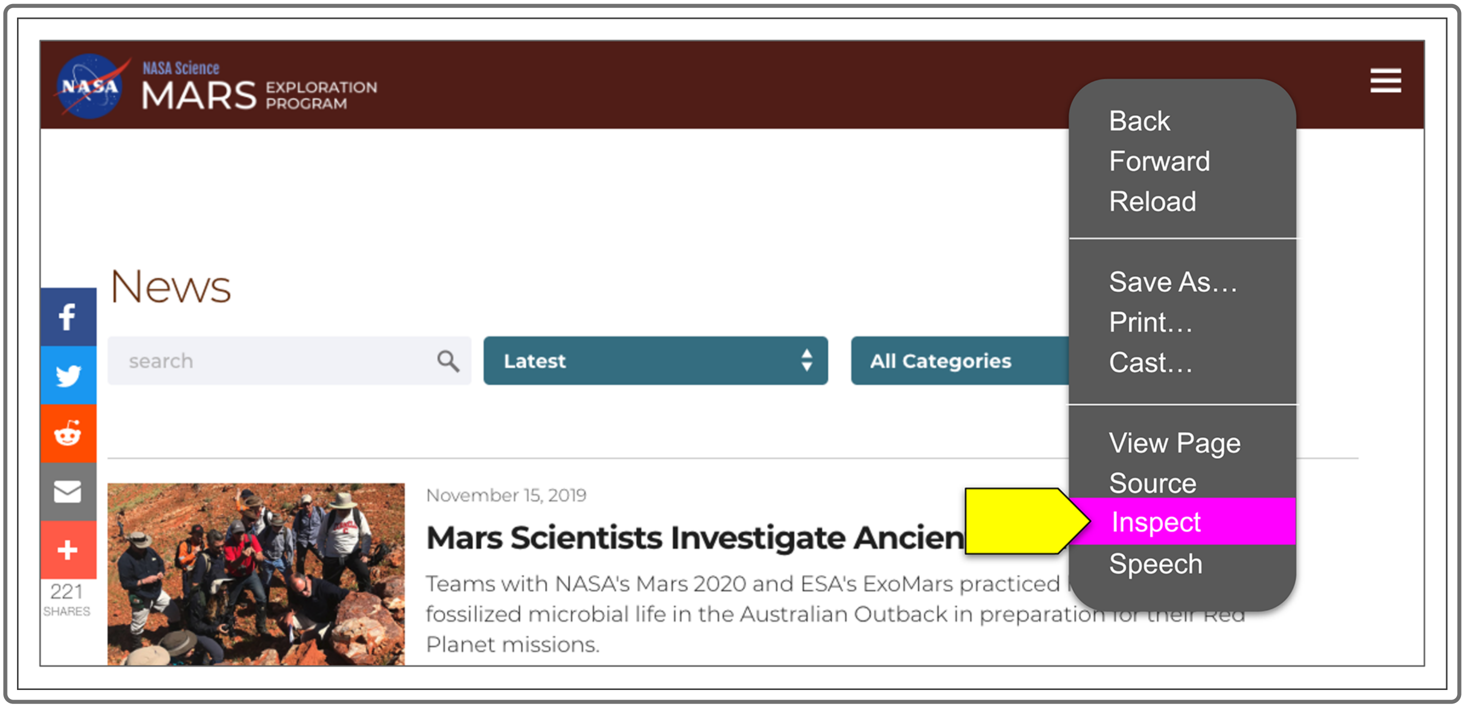
In this section, you'll learn how to use CSS selectors to efficiently find elements in Chrome DevTools.

Using Chrome DevTools

As you scrape websites, you'll often find that the HTML element you want is contained within another element—or even buried inside layers of elements. DevTools makes it easier to identify such an element and its CSS selectors. We can then use those selectors to scrape the data from the element.

Start by opening the News – NASA Mars Exploration (https://mars.nasa.gov/news/?page=0&per_page=40&order=publish_date+desc%2Ccreated_at+desc&search=&category=19%2C165%2C184%2C204&blank_scope=Latest) website in a new browser window. We can initially observe article titles and a sentence that describes each article.

Next, open DevTools by right-clicking anywhere on the page and then clicking Inspect.



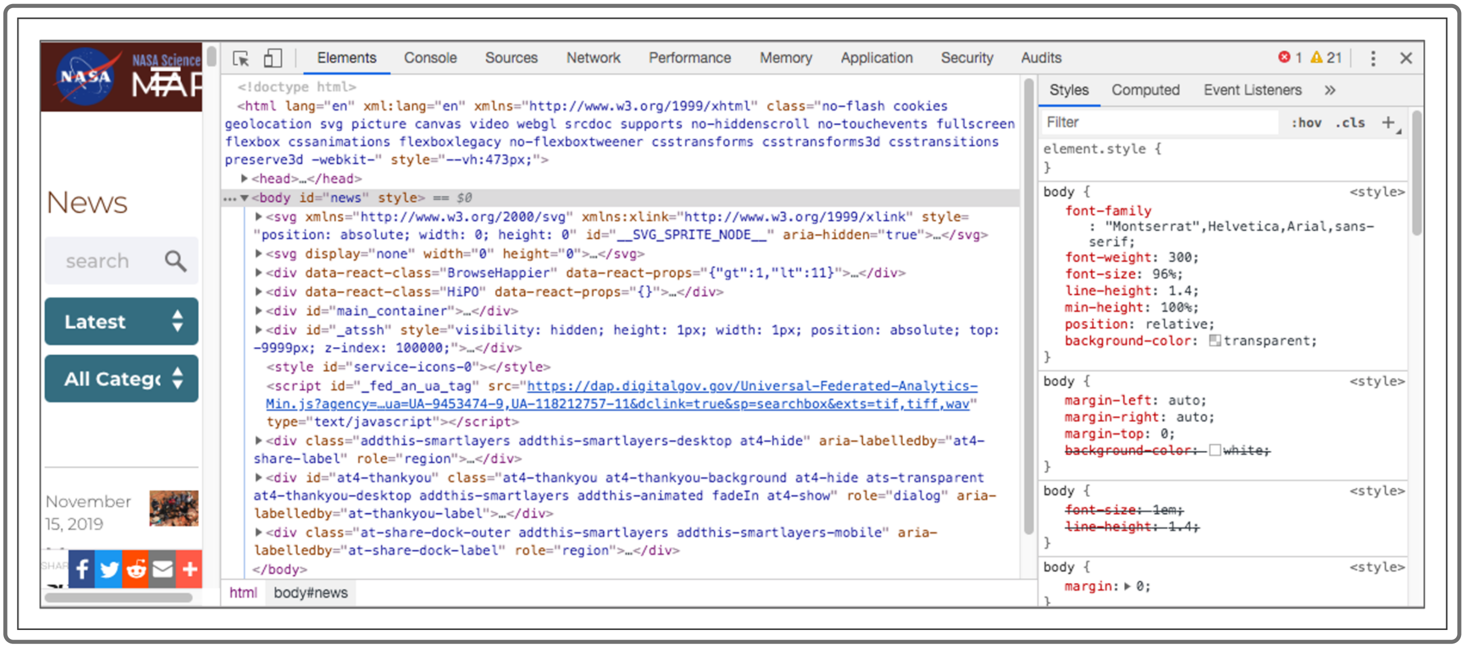
News

November 15, 2019

Mars Scientists Investigate Ancien

Teams with NASA's Mars 2020 and ESA's ExoMars practiced fossilized microbial life in the Australian Outback in preparation for their Red Planet missions.

A new panel opens. This panel is docked to the webpage but has a different job than the webpage.



```
<!doctype html>
<html lang="en" xml:lang="en" xmlns="http://www.w3.org/1999/xhtml" class="no-flash cookies
geolocation svg picture canvas video webgl srcdoc supports no-hiddenscroll no-touchevents fullscreen
flexbox cssanimations flexboxlegacy no-flexboxtweener csstransforms csstransforms3d csstransitions
preserve3d -webkit-" style="--vh:473px;">
 ▸<head>…</head>
…▾<body id="news" style> == $0
   ▸<svg xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink" style=
   "position: absolute; width: 0; height: 0" id="__SVG_SPRITE_NODE__" aria-hidden="true">…</svg>
   ▸<svg display="none" width="0" height="0">…</svg>
   ▸<div data-react-class="BrowseHappier" data-react-props="{"gt":1,"lt":11}">…</div>
   ▸<div data-react-class="HiPO" data-react-props="{}">…</div>
   ▸<div id="main_container">…</div>
   ▸<div id="_atssh" style="visibility: hidden; height: 1px; width: 1px; position: absolute; top:
   -9999px; z-index: 100000;">…</div>
    <style id="service-icons-0"></style>
    <script id="_fed_an_ua_tag" src="https://dap.digitalgov.gov/Universal-Federated-Analytics-
    Min.js?agency=…ua=UA-9453474-9,UA-118212757-11&dclink=true&sp=searchbox&exts=tif,tiff,wav"
    type="text/javascript"></script>
   ▸<div class="addthis-smartlayers addthis-smartlayers-desktop at4-hide" aria-labelledby="at4-
   share-label" role="region">…</div>
   ▸<div id="at4-thankyou" class="at4-thankyou at4-thankyou-background at4-hide ats-transparent
   at4-thankyou-desktop addthis-smartlayers addthis-animated fadeIn at4-show" role="dialog" aria-
   labelledby="at-thankyou-label">…</div>
   ▸<div class="at-share-dock-outer addthis-smartlayers addthis-smartlayers-mobile" aria-
   labelledby="at-share-dock-label" role="region">…</div>
  </body>
```

```
element.style {
}
body {                                          <style>
   font-family
     : "Montserrat",Helvetica,Arial,sans-
     serif;
   font-weight: 300;
   font-size: 96%;
   line-height: 1.4;
   min-height: 100%;
   position: relative;
   background-color: ▨transparent;
}
body {                                          <style>
   margin-left: auto;
   margin-right: auto;
   margin-top: 0;
   ~~background-color: □white;~~
}
body {                                          <style>
   ~~font-size: 1em;~~
   ~~line-height: 1.4;~~
}
body {                                          <style>
   margin: ▸ 0;
}
```

As we can observe in the panel, this site has a lot going on. What's the overall structure of the site? Well, the line that begins with `<html lang="en">` should seem familiar. And, so should the `<head />` and `<body />` tags. But, what's all the other stuff? And, what's the stuff that appears inside the familiar tags?

Drag the HTML tags to the correct positions in the code.

```
<          id="news" style="">
  <          display="none" width="0" height="0">
    <          id="circle_plus" height="30" viewBox="0 0 30 30" width="30">
      <          fill-rule="evenodd" transform="translate(1 1)">
        <          cx="14" cy="14" ...><          >
        <          class="the_plus" d="m18.856 ...><          >
      <          >
    <          >
   <          >
<          >
```

g circle /circle /symbol path svg body /body

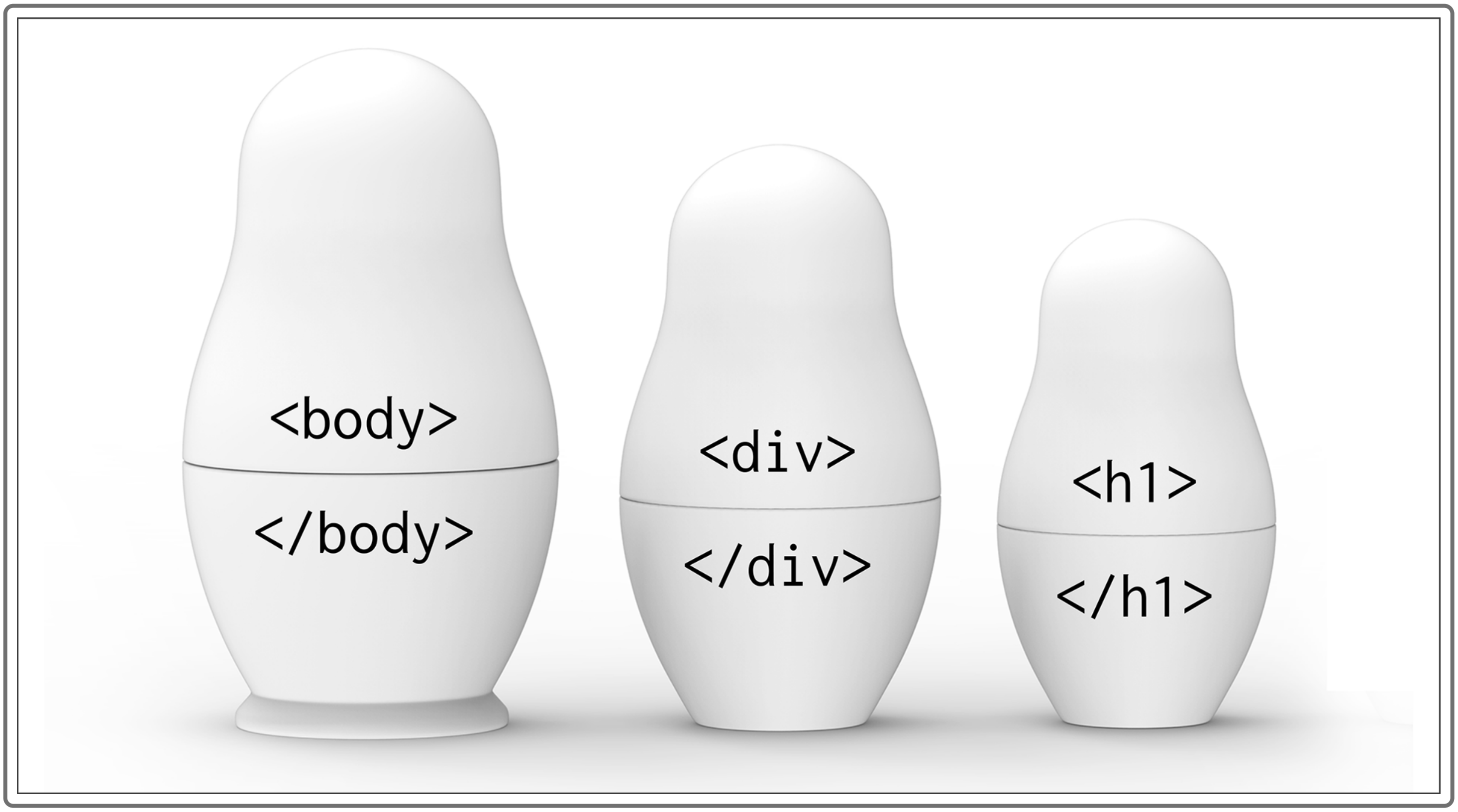
/svg /path /g symbol

Check Answer

Finish ▶

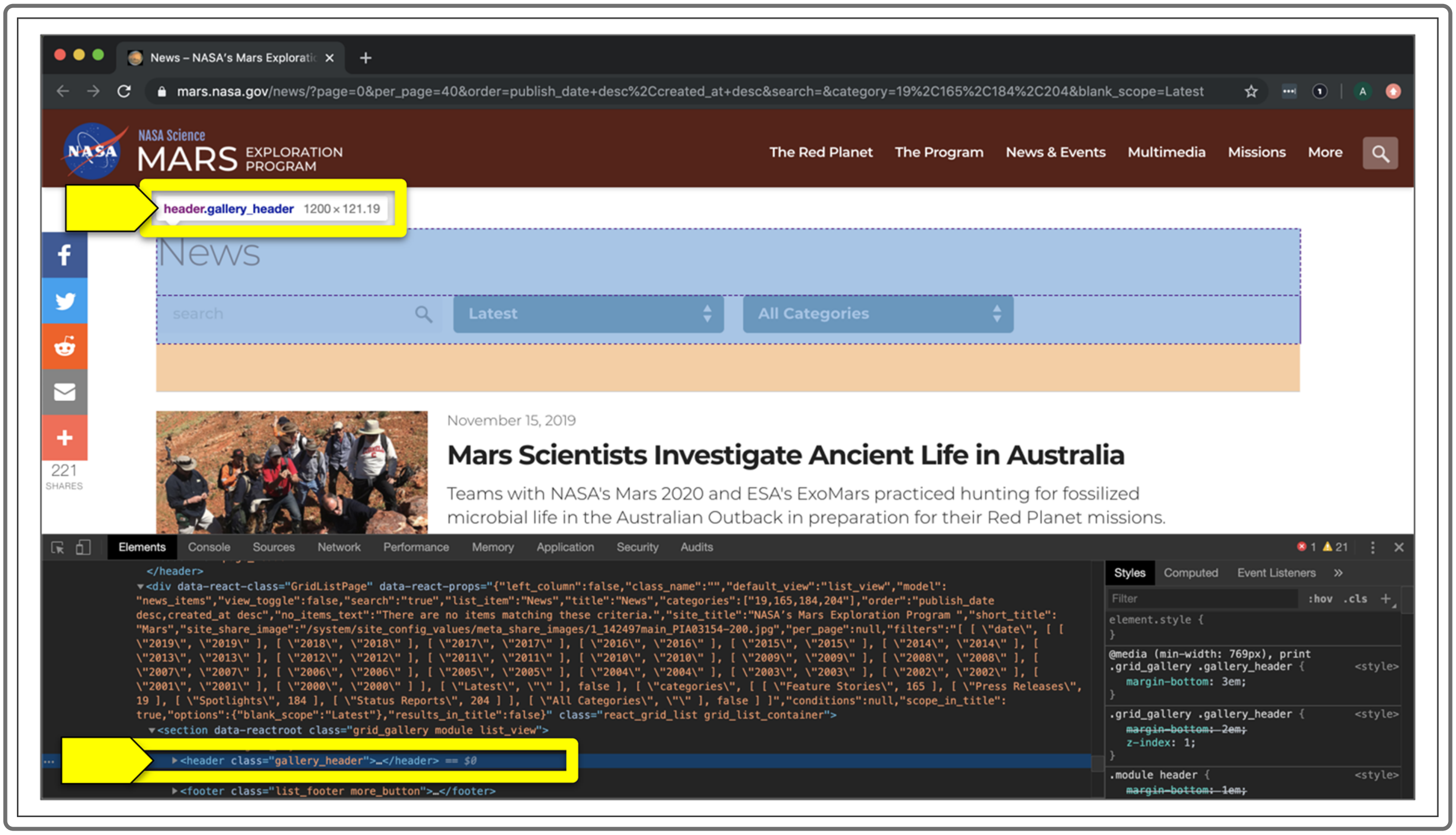
Let's break down the structure of this site a bit. Remember that an HTML element can contain other elements. For example, the `<body />` tag is a container of every element that appears on the webpage, such as the header and paragraph elements. And, other containers can also exist inside that `<body />` tag. These containers are **nested**, much like in a nesting doll. That is, one element can contain another element which, in turn, can contain another element, and so on. Furthermore, multiple levels of nesting can exist, depending on how elaborate the website is. In the case of this website (and most websites), the other containers inside the body consist of `<div />` tags.

The following image shows that each container is nested within another. In this case, an `<h1 />` element is nested within a `<div >` container which, in turn, is nested within the `<body />` container:



DevTools has another feature. Specifically, when we hover over any part of the code, the connected visual gets highlighted on the webpage. This helps by showing us which piece of code is tied to which feature of the webpage.

This webpage contains lots of custom code. So instead of scrolling through all of it to find a certain element, we'll just search for it in DevTools. To do so, press Ctrl+F (on Windows) or Command+F (on macOS) to display the Find box. In the box, enter "gallery_header" (without the quotation marks), and then press Enter. It may return more than one result, so you may have to press Enter to advance until the "header class="gallery_header" line is selected, and then hover over it. Make sure that the "header class="gallery_header" line is selected, and then hover over it. The header section of the page, which consists of the title and its container element, gets highlighted.



November 15, 2019

Mars Scientists Investigate Ancient Life in Australia

Teams with NASA's Mars 2020 and ESA's ExoMars practiced hunting for fossilized microbial life in the Australian Outback in preparation for their Red Planet missions.

```
    </header>
  ▼<div data-react-class="GridListPage" data-react-props="{"left_column":false,"class_name":"","default_view":"list_view","model":
  "news_items","view_toggle":false,"search":"true","list_item":"News","title":"News","categories":["19,165,184,204"],"order":"publish_date
  desc,created_at desc","no_items_text":"There are no items matching these criteria.","site_title":"NASA's Mars Exploration Program ","short_title":
  "Mars","site_share_image":"/system/site_config_values/meta_share_images/1_142497main_PIA03154-200.jpg","per_page":null,"filters":"[ [ \"date\", [ [
  \"2019\", \"2019\" ], [ \"2018\", \"2018\" ], [ \"2017\", \"2017\" ], [ \"2016\", \"2016\" ], [ \"2015\", \"2015\" ], [ \"2014\", \"2014\" ], [
  \"2013\", \"2013\" ], [ \"2012\", \"2012\" ], [ \"2011\", \"2011\" ], [ \"2010\", \"2010\" ], [ \"2009\", \"2009\" ], [ \"2008\", \"2008\" ], [
  \"2007\", \"2007\" ], [ \"2006\", \"2006\" ], [ \"2005\", \"2005\" ], [ \"2004\", \"2004\" ], [ \"2003\", \"2003\" ], [ \"2002\", \"2002\" ], [
  \"2001\", \"2001\" ], [ \"2000\", \"2000\" ] ], [ \"Latest\", \"\" ], false ], [ \"categories\", [ [ \"Feature Stories\", 165 ], [ \"Press Releases\",
  19 ], [ \"Spotlights\", 184 ], [ \"Status Reports\", 204 ] ], [ \"All Categories\", \"\" ], false ] ]","conditions":null,"scope_in_title":
  true,"options":{"blank_scope":"Latest"},"results_in_title":false}" class="react_grid_list grid_list_container">
    ▼<section data-reactroot class="grid_gallery module list_view">
...     ▶<header class="gallery_header">…</header> == $0
        ▶<footer class="list_footer more_button">…</footer>
```

```
element.style {
}
@media (min-width: 769px), print
.grid_gallery .gallery_header {            <style>
   margin-bottom: 3em;
}
.grid_gallery .gallery_header {            <style>
   ~~margin-bottom: 2em;~~
   z-index: 1;
}
.module header {                           <style>
   ~~margin-bottom: 1em;~~
```

Next, hover over the following line of code, which is `<h2 class="module_title">News</h2>`. (If the header doesn't display the nested contents, click its arrow to expand it.)

The highlighted portion of the webpage becomes smaller. That's because we hovered over an element that's nested inside a container instead of hovering over the full container.

This is a splendid way to pinpoint where on the website we want our web-scraping code to extract data from. We can't tell the code to just extract a division or a header, though. That's because many of these might exist on the webpage, when we want only one. And, that's where the `class` and `id` attributes come into play.