"Tourist Bureau"

Workbook 4's Workshop

Project Description

You will build a website for the local visitor's bureau that will help tourists explore all the things to do in the area. If they find an excursion they like, they can buy "e-tickets" directly from the site.

They want the following features:

- A home page with informational text/images promoting the town and navigation links to the other pages***. It needs a professional, but minimalistic appearance. Their budget for this project is tiny!***
- A page that lets tourists search for and read about activities in the area. If they like an activity, they can buy an e-ticket for it right there.
- A page that lets tourists learn about popular local hikes. (***This page is a bonus page. Make sure to do a good job on the activities page before attempting this page.)***
- A page that lets tourists watch a video about the area. You can use any YouTube video (or other source) you find that seems reasonable. (***This page is a bonus page. Make sure to do a good job on the other pages before attempting this page.)***

Accompanying this workshop write-up is a ZIP file containing data and images you will need for the lab. It will include:

1) JavaScript files that define an array of activity categories, an array of activity details, and an array of hike information.
2) Images for the hiking page that you may resize, edit, or replace as needed.

See the implementation details section of each page for more information.

Details about the Activities Page (REQUIRED)

The page will feature a dropdown list that contains the types (categories) of activities offered by vendors in our community. They include: Adventures, Museums, Wine Tastings, etc. Make sure a "Select one" option appears at the top of the list and is the one pre-selected.

When the user selects an activity category, a dropdown will display the names of activities that vendors offer that match the selected category. Make sure a "Select one" option appears at the top of the list and is the one pre-selected.

When the user selects an activity, details of that activity (id, name, description, location, price, etc) will be displayed on the page. If the price is more than $0.00, the page will also display a form with fields that can be used to buy an e-ticket. The form should have (**only**) the following fields:

- the number of tickets needed
- a credit card number
- an email address
- a purchase (submit) button
- a reset button

When the user clicks the purchase button, a message will appear below it stating something to the effect:

> "Your credit card has been charged $(amount) for (number-of- tickets) to (adventure-name). A confirmation email has been sent to (email)."

Of course, in real life, the form's submit event handler would send the information to a server for processing.

IMPLEMENTATION DETAILS: The two arrays provided for this page will resemble:

```
let categories = ["Adventures", "Arts & Crafts", ...];

let activities = [
   {
      category: "Adventures",
      id: "A101",
      name: "Hot Air Balloon Ride",
      description: "Enjoy a lovely hot air balloon ride
over the valley at sunrise.  Call 800-555-1212 to reserve
a date/time after you complete your e-ticket purchase.",
      location: "121 S. Main Street",
      price: 365.00
   },
   /* more here */
];
```

You will also need to design your JavaScript for this page where you have ***at least*** three helper functions. Suggestions for helper functions include:

1. Hiding or showing sections of the page
2. Displaying the activity details

Details about the Hiking Page (BONUS)

The page will feature a dropdown list of local hikes loaded from the data. Make sure a "Select one" option appears in the dropdown list. When the user selects a hike, details about that hike will be displayed along with two images (one taken along the hike and another showing a trail map). Sample images are included in the data files.

To see how to change the source for an image programmatically using JavaScript, see the following:

- Changing `src`: `https://www.w3schools.com/jsref/prop_img_src.asp`
- Changing `alt`: `https://www.w3schools.com/jsref/prop_img_alt.asp`

IMPLEMENTATION DETAILS: The array provided to you for this page will resemble:

```
let hikes = [
   {
      id: "H101",
      name: "Some Hike Name",
      description: "Some description of the hike.",
      scenicImage: "hike_1.png",
      trailMapImage: "hike_2.png",
      length: "2 miles",
      difficulty: 2
   },
   /* more here */
];
```

What Makes a Good Workshop Project?

You should:

- build a consistent look-and-feel throughout the site with intuitive navigation
- implement at least the required pages

- have a responsive user interface

You should adhere to best practices such as:

- have a good directory structure (ex: `css`, `images` and `scripts` folders)
- include Bootstrap from a CDN
- have good file naming conventions (ex: lowercase file names with no spaces)
- have well-formatted HTML, CSS, and JavaScript (indentions, blank lines, etc.)
- use good names for your HTML elements and JavaScript variables/functions
- use HTML, CSS, and JavaScript comments effectively

Make sure that:

- you use the ESLint tool to ensure you've written good JavaScript!
- you use validators to ensure you have no HTML or CSS errors!
- there are no JavaScript errors at run time (check the Console tab in the browser)

Build a **PUBLIC** GitHub Repo for your code.

- Use an appropriate branch structure and have a commit history with meaningful comments
- Include a README.md file that describes your project and includes screen shots of 1) your home page 2) EACH of the pages you implemented that shows several interesting views of each page (labeled as to why they are interesting)
- In your README, ***make sure to have a section*** where you point out cool features and/or things you are excited about that you implemented in your site
- Make sure it is on the first page of your GitHub project!

Things we look for as we play with your website include:

- Does each page function as intended?
  - Do the right activities show up when a category is picked?
  - Does the second list clear if "Select one" is picked in the first dropdown list?
  - Do the activity details clear if "Select one" is picked in the activities dropdown?
- Does a "purchase tickets" form appear if the activity selected isn't free?
  - Do you require user inputs by using the HTML `required` field validation attribute?
  - Are we unable to enter bad data because input is restricted using validation attributes? (ex: `<input type="number">` or `<input type="email">` ?)
  - Do you use a submit button to trigger processing?
  - Do you display monetary values with two digits to the right of the decimal point (and maybe a dollar sign in front of the number if that makes sense)?
  - Do you have a reset button to clear the fields?
  - Does the reset button also clear the "Your credit card has been charged…" message?
- On the hikes page (if you implement it), do the images show correctly?
  - Do the images and hike info clear if the "Select one" option is picked?
- On the video page (if you implement it), does the view play?