"Vacation Resort"
Workbook 3's Workshop

Project Description

You will build a website for a vacation resort that will help tourists plan their stay. They have tossed around ideas of what pages they want, including:

- A home page with informational text about the resort, images, and navigation links
- A page that helps guests plan their _overnight stay_ based on parameters the user selects
- A page that helps guests plan their _dining_
- A page that helps guests plan their _activities_ while at the resort

In this workshop, you will use **_submit_** buttons on forms and handle the form's `submit` event instead of a button's `click` event. This will allow you to use HTML form validation features.

The only REQUIRED pages are the first two. But you may surprise yourself and finish early. If so, try the dining page. When that's done, if you still have energy and time, try the activities page!

Details about the Overnight Cost Estimator Page (REQUIRED PAGE)

This page allows a user to determine how much a hotel room will cost at the resort on a given day for a given room type.

To get a price quote, users must:
- enter their name and email
- select a check-in date
- specify a number of nights for the visit (1 - 28 days)
- select a room type (use radio buttons that specify: Queen, King, 2-Bedroom Suite)
- specify the number of adults (1 - 4) and number of children (0 - 4) that will stay in the room being reserved
- select whether they will use any discounts (use radio buttons that specify: None, AAA/Senior (10%), and Military (20%))

The resort's price list is currently is determined by the check-in date:

Room Type	Max Occupancy	Jun - Aug Per Night	Rest of Year Rates Per Night
Queen	5 (inc rollaway)	250.00	150.00
King	2	250.00	150.00
2-Bedroom Suite	6 (inc sleeper sofa)	350.00	210.00

You price a stay using the in-season and out-of-season rates based on their _check-in_ date. For example, if someone checks in to a Queen room on 3/20 (March) the rate is $150/night, but if they check in on 6/20 (June), the rate is $250/night. NOTE: The same rate is applied throughout the entire stay and is solely based on check-in date.

In this program, you should implement a function named `getRoomRate()` that is passed in the check-in date and room type. It will return a value like $250 or $150 that represents the nightly rate for that room.

> DEVELOPER's TRICK: To make pricing a room easy, start by always returning 150.00 from the function and get the rest of the page working. Then go back in that function and put the correct logic for the specified inputs.

If the user has AAA or requests a senior discount, we take 10% off the room rate. If they are a veteran or request military pricing, we take 20% off the room rate.

Taxes are 12% of the after-discount room cost.

When the "estimate stay cost" button is clicked, the page should display:
- the original room cost
- the discount, if any
- the discounted room cost
- the tax
- the total cost of the stay

IMPORTANT: _**If the room type will not support the number of guests they have entered**_, display a message and DO NOT calculate or display the cost of their stay. For example:

```
const messageDiv = document.getElementById("messageDiv");
messageDiv.innerHTML =
    "The room you selected will not hold your party".
```

BONUS FEATURE: Display a "confirmation number" on the page along with the price estimate that is made up of:

> the first three characters of their name
> a hyphen
> the month and year they are checking in
> a hyphen
> the number of days of the stay
> a colon
> the number of adults on the reservation
> a colon
> the number of kids on the reservation

For example: DAN-102021-7:2:0

Details about the Dining Options Page (BONUS PAGE)

The second page will provide information about the restaurants and dining options available at the resort. Guests have two dining options at the resort:

- Pay as you go
- All-inclusive

Your page should have two radio buttons at the top that allow the user to select one of the dining options. It will toggle (hide/show) what is displayed below the radio buttons.

For the **pay as you go option**, it will show the names of a couple of restaurants, a brief description, and a small food image next to each. (Note: You can use lorem ipsum or bacon ipsum text to describe the restaurants.)

For the **all-inclusive option**, it will show a price chart that resembles the following and a message that the all-inclusive option can be added to their reservation at check-in.

Plan	Adult	Child
Basic	$62.50/day	$49.99/day
Premium (incl drinks)	$80.00/day	$49.99/day

BONUS FEATURE: In the all-inclusive option, let the user enter their confirmation number (from the first page) and calculate the expected dining cost (by parsing out the number of days of stay, the number of adults, and the number of children) and display a message resembling:

>   "The all-inclusive option has been added to your reservation. The price of your stay has increased by $xxx.xx based on your selections here."

Details about the Activities Page (BONUS PAGE)

This page provide a listbox that lets the user select a category of activities they are interested.
Categories include:
- Arts/Crafts
- Adventure
- Museums/Culture

When the user selects a category, the page will display a list of activities (including a description and price) for items in that category. You can show these activities in an HTML table, Bootstrap cards, or other "cool" HTML solution.

You will need to compile your own list of activities, descriptions, and prices. But minimally, you should include:

Category	Title	Description	Cost
Arts/Crafts	Candle Making	Spent two hours learning to make scented candles. All supplies are included.	7.99
...			
Adventure	Ziplining the Canyon	Spend 2 hours ziplining across canyons and through treetops.	119.99
...			
Museums/Culture	Amazing Science	Visit our own science museum with its world-class exhibit of magician tricks.	9.99
...			

IMPLEMENTATION HINTS: You can create a listbox by using a `<select>` element that has a `size` attribute. Rather than a dropdown, it will show the number of elements that match the size and provide a scroll bar if the number of elements exceeds the size.

```
<select id="activityTypesSelect" size="4">
...
</select>
```

You can detect a change in a listbox by handling the `select` element's `onchange` event. To get the selected element in the `select`, look here: `https://ricardometring.com/getting-the-value-of-a-select-in-javascript`

The easiest way to show the activities of a specific category to use the magic of hide/show. Have one div for the items in each category.   Then hide or show each div based on the user's selection.

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 structures (ex: `css`, `images` and `scripts` folders)
- include Bootstrap and jQuery 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 inputs and correct outputs -- showing at least two different outputs from each, and 3) a mobile view of one page

- 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

Things we look for include as we play with your website:
- Does each page functions as intended?
- Is focus automatically set to the first input field if the page has a form?
- 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="date">` ?)
- Do you use min and/or max in `<input type="number">` fields to help restrict user input?
- Are you using `readonly` on output form fields (if appropriate)?
- 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 use a ***submit*** button to trigger processing when you need a button?
- Do you have a reset button on your forms?