

Prog 1247 - Web Foundations

Assignment 3: Building a Web Form for Ordering Event Tickets

Introduction

This assignment is yet another next step in the further development of the New Year Central website. You are to build a web form for ordering tickets for an event.

- The source code, which is the starting point for this assignment, can be downloaded from eConestoga: Evaluations → Assignments → Assignment #3 → **Assignment3-Source-Files**
- The instructions for the assignment, below, include:
 - an image of how the page should appear in browser, along with
 - task specifications which you can complete in whichever sequence you prefer.
- The tasks are detailed enough for you to complete them. However, you'll need to use your best judgment on how to code the details. To do that, write the code in the way that you think is best, based on the skills that were presented in the class. The goal is to use best HTML5 and CSS coding practices.
- Make sure to follow the design requirements as well: once completed **your form page should display as per the screenshot** included below.
- Recommended resources to use as support and help with the assignment:
The lesson material and exercises covering web forms.

Assessment

- You will be graded by the rubric found at the end of this document.
- This is an independent work assignment - you may not work with others while working on this assignment or collaborate or share your answers, code or anything else.
- You may use lecture material and in-class exercises.
- You may use the web, for generalized web development practices research.
- You may not use copied code from other people, the web or any other source.
- It is your responsibility to read and understand the Conestoga College Student Responsibility Policy

How to Submit

Note: a 15% penalty will be applied if any of the submission requirements listed below are not met.

For this assignment there are two parts when it comes to its submission:

- **Step 1: Upload and publish your web page on GitHub.com**

1. Create a free account on GitHub.com
2. Upload your completed assignment files under your GitHub account
3. Issue a GitHub page to display your work


For instructions on how to use GitHub, refer to the video tutorial included in Week 10 lesson slides and the associated recording.

- **Step 2: Upload your assignment files and the link to your GitHub page to the Assignment 3 Dropbox on eConestoga**

1. Save a copy of your tickets_order_form.html and main.css files as .txt files and submit these two .txt files to the Assignment 3 Dropbox on eConestoga.
2. Also submit the link to your GitHub page, as a comment. Note that I will use the link to review your web page and the code behind, and to mark your assignment, so make sure to test the link and confirm that it is working before submitting it.

Instructions for Creating the Ticket Order Form

You will use HTML5 and CSS to build a web form to be used for placing orders for New Year events tickets. **When you are done, the page should look like this:**



New Year Central

For the child in all of us!

[Home](#)[About Us](#)[Our Products](#)[Event Tickets](#)[Contact Us](#)

Products of The Week...

- Natural Holly Berry Garland, 5f - \$74.99
- Artificial Red Berry and White Edge Holly Leaf Christmas Garland, 5.5f - \$38.99
- Christmas Berries Red Stems Evergreen Pine Branches, 8pcs - \$15.99

Order Event Tickets

Member Information

E-Mail:

First Name:

Last Name:

Address:

City:

Province:

2-character code

Postal Code:

6 characters

Phone Number:

999-999-9999

Ordering Information

Event:

Cookie Decoration Contest ▾

Order Type:

Member Package ▾

Number of Tickets:

Number of single tickets

Payment Method

☐ Bill Me ☐ Credit Card

Credit Card Information

Card Type:

Visa ▾

Card Number:

16 digits

Expiration Date:

January ▾

2021 ▾

Submit Your Order

Submit

Reset Fields

© 2021 New Year Central. Created by Johnny White.

Task specifications:

You will be creating the order form in **tickets_order_form.html web page**, which is found in Assignment-3-Source-Files folder.

1. **Code the form element:**

- Use 'order_form' as the value for the form's name attribute
- Use 'post' as the form submission method
- When you click on the Submit button, a web page called place_order.html is displayed (hint: use the form's action attribute). The place_order.html file is located in the same folder as your tickets order page.

2. **Add fields/controls** to the form as per the screenshot provided above:

- Make sure to use label elements for displaying the text beside the input fields.
- The Event select control should have the following choices:
 - Cookie Decoration Contest
 - Tree Decoration Contest
 - Gift of Light Ride-Along
 - Breakfast with Santa
- The Order Type select control should have the following choices:
 - Member Package
 - Donor Package
 - Single Tickets
- The Card Type select control should have the following choices:
 - Visa
 - Master Card
 - Discover
- Code the two radio buttons so that:
 - They are part of the same radio group (ie. clicking on one of them deselects the other one)
 - User can activate them by clicking on the labels
- The two select controls used for Expiration Date are to have the following choices
 - The month list should have values from January through December
 - The year list should have values from 2021 through 2027
- Use fieldset and legend elements to group the form elements as shown

- Set the autofocus on the Email field.

3. **Add validation for form data entry** using the available HTML5 and CSS3 data validation 'tools':

- If you haven't already, make sure that you use the appropriate HTML5 elements for the Email, Phone Number and Number of Tickets fields.
- Code (only) the following fields on the form as required using the appropriate HTML5 attribute:
 - Email, First Name, Last Name, City, Province, Postal Code, Card Number
- Use the appropriate HTML5 attribute to specify regular expressions to be used to validate the data entered so that it matches the following patterns:
 - Province: expecting a 2-letter code
 - Postal code: expecting exactly 6 alphanumeric characters
 - Phone Number: expecting the value to be provided in '999-999-9999' format
 - Number of tickets: expecting a number
 - Card Number: expecting a 16-digit entry

NOTE: Look up regular expressions you'll need for this task in the lesson slides (week 10) and/or in the exercise we have done in the class.

- When a user hovers over each of the fields validated with a regular expression, a message should be displayed informing the user of what the accepted data format is, for that particular field. (As an example, see the message shown on mouse over the Postal Code field, on the form screen shot.)
- Use the placeholders for the form fields as shown in the screen shot.

4. **Style the form and its elements as per the design**, by modifying/adding CSS styles in main.css:

- Use margin to add top and bottom spacing for fieldsets elements as needed
- Use margin to add bottom spacing for legend elements
- Use bold and grey colour for legend text
- Set font-size as follows:
 - for legend elements: 85%
 - for label, input and select elements: 90%
- Style the labels and input fields so that they appear as per the design:

- the labels display on the left, input fields on the right side
 - the label text is aligned to the right
 - input fields have same width (15 em) and are all aligned
 - use margins on input fields to add left and bottom spacing as needed
- Add CSS styles so that the required fields have a 2-pixel orange border, and the valid fields have a 1-pixel black border.
 - Adjust the CSS so that the radio buttons are displayed side by side, and their respective labels on their right, as shown.

Hint: To achieve this you can code the id attribute for the fieldset element that is grouping the buttons (id='payment_method'), and then add the following styles to main.css:

```
#payment_method label {

    /* here add the style that will remove the float for the labels beside the
    radio buttons, so they appear on the right side of the buttons themselves
    */

}

input[type="radio"] {

    /*
    - here (re)set the width for the radio button controls to 'auto', and
    - set the left margin for the radio buttons to add space as per the design
    */

}
```

- Format month and year lists as shown.
Hint: you can use id selectors to set the width of the month control to 7 em and the width of the year control to 5 em.
- Have the Submit and Reset Fields button 10 em wide.

Marking Rubric

	1	2	3	4	5
Completeness per Requirements (x2)	Minimal completeness to requirements. Scoring at this level limits the overall mark to 55%.	Many major requirements not implemented. Scoring at this level limits the overall mark to 65%.	Multiple missing minor items or a major miss. Scoring at this level limits the overall mark to 75%.	Excellent implementation of requirements; any errors or omissions are few and minor. Scoring at this level limits the overall mark to 85%.	Flawless execution of all requirements.
Technical Accuracy (x2)	Code does not render. Scoring at this level limits overall code marks at 55%.	Several major errors or omissions in logic, semantics or technical code affecting correct operation. Scoring at this level limits the overall mark to 65%.	A major or multiple minor errors in technical logic or semantics and code accuracy. Scoring at this level limits the overall mark to 75%.	Highly accurate coding, any logic or semantic error is limited and minor. Scoring at this level limits the overall mark to 85%.	All features of level 4 but with flawless accuracy and efficiency.
Documentation & Style	Missing sufficient comments and formatting, as well as extensive issues with code style. Scoring at this level limits the overall mark to 55%.	Missing both sufficient comments and formatting, or extensive issues with code style. Scoring at this level limits the overall mark to 65%.	Several minor or one major issue in code style, missing sufficient comments have non-minor issues with clarity. Scoring at this level limits the overall mark to 75%.	Excellent documentation, including sufficient comments. Any miss in consistency to best practices of code style is few and minor. Scoring at this level limits the overall mark to 85%.	All features of level 4, with further excellency shown in the professionalism of the documentation and consistency of style. Exceedingly polished appearance.