**Making justification**

I am aiming for Credit for this unit. I'll be adding my reflections to my portfolio in the coming weeks. I will follow the assessment guide and try my best to complete the weekly practice tasks and not exceed the weekly word count.

**Week 1 – Introduction HTML and CSS**

**Reflections**

**HTML** is a programming language (Hypertext Markup Language). It can include a variety of features in addition to text, such as images, music, links, programmer, and so on. HTML is a language that allows a human and a browser to communicate. It uses multiple tags to categories the parts of a page and inform the browser how to display the content.

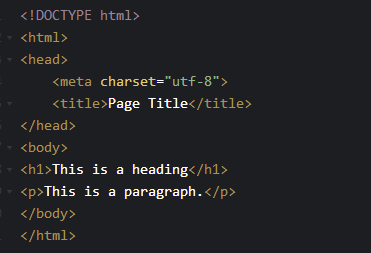
**CSS** is used to style HTML content, such as beautification, and is simply responsible for the appearance and layout of the elements on an HTML page.

**JavaScript** is a scripting language that can be used on both the client and server sides. JavaScript is mostly used to make static HTML interactive by extending document interaction (such as form submission animations, pop-ups, etc.)

To summaries, if HTML is a human body, CSS is clothing, and JavaScript is enhanced human interaction.

**Lab Exercise Answers**

**Task 1 – Create an HTML Page**



This is an example of a simple html document and its output HTML code



Common html Tags

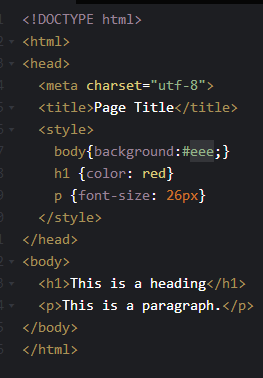
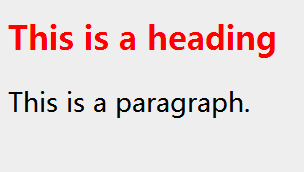
|  |  |
| --- | --- |
| HTML Tag | Description |
| <html> … </html> | The root element of the Page and contains all |
| <head> … </head> | Contain metadata such as title and what files  to include |
| <title> … </title> | the title of the page |
| <body> … </body> | Where all the actual content of the website goes |
| <h1> … </h1> to <h6> … </h6> | HTML 标题（Heading）是通过<h1> - <h6> 标签来定义的 |
| <p> … </p> | HTML 段落是通过标签 <p> 来定义的。 |
| <a>...</a> | HTML 链接是通过标签 <a> 来定义的。 |
| <img/> | HTML 图像是通过标签 <img> 来定义的. |
| <table> … </table> | 表格由 <table> 标签来定义。每个表格均有若干行（由 <tr> 标签定义），每行被分割为若干单元格（由 <td> 标签定义） |
| <ul> <li>...</li> </ul> | 无序列表是一个项目的列表，此列项目使用粗体圆点（典型的小黑圆圈）进行标记。 |
| <ol> <li>...</li> </ol> | 有序列表也是一列项目，列表项目使用数字进行标记。 |

**Reflections:**

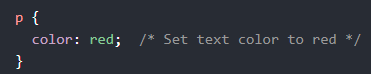
In this section, I learned how to create HTML files and the basic structure of HTML files. Understand the common HTML tags used in making pages. Understand the role of HTML, CSS, JavaScript in the website. HTML is like the frame of the house, CSS is like the decoration of the house. JavaScript mainly implements some of the interactions within a web page.

**Task 2 – Add CSS to your HTML page**

Add internal, inline, and external

Example from w3schools



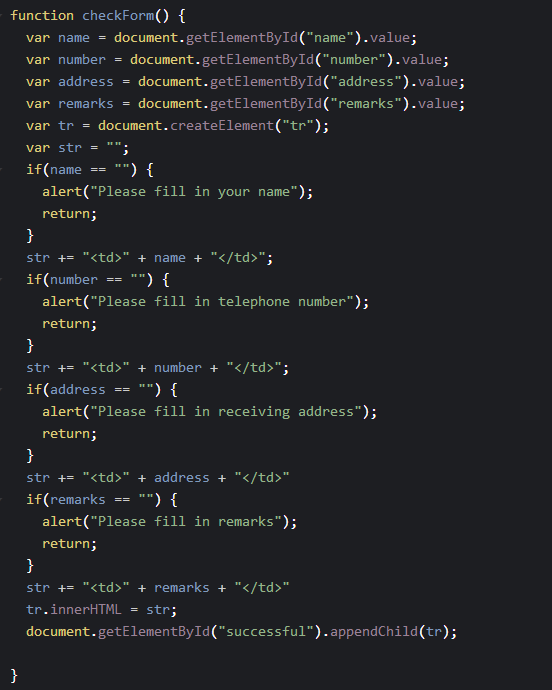
What I learned

Picture 10

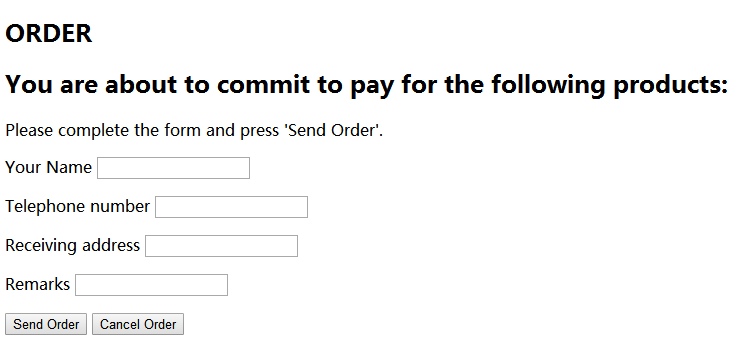
**Reflections:**

There are three ways to introduce CSS to a page: internal, external, and inline. You can set font color, background picture, style color, set the container inside and outside margin, indent distance, etc.. as well as set the animation of the web page and so on all styles.

Task 3 – Adding JavaScript



This is an example of a simple html document and its output HTML code



**Reflections:**

JavaScript can verify that form elements are filled in, create new tags through DOM manipulation, and append them to their desired locations.

Task 4 – Vue.js Framework(homework)





**Reflections:**

The V-for directive renders a list based on an array. The V-for directive requires the special syntax of the form Item in Items, where items are the source data array and item is the alias of the array element being iterated over.

# Week 2 -Responsive Web UX/UI

**Reflections:**

Responsive layout is a concept that, in short, means that a website can accommodate multiple terminals and provide a more comfortable interface and better user experience for users of different terminals, rather than making a specific version for each terminal. The concept was born as a solution to mobile Internet browsing.

**Task1**

Adaptive Design: The design adapts based on the viewport width.

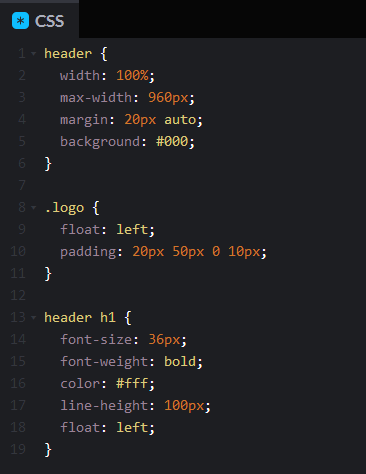
<meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0" />

[Responsive Layout @media screen and ( max-width: pixel value ) {}](https://www.cnblogs.com/shen9449/p/8151434.html)

I first defined a fixed width for the standard browser (if the standard browser resolution is 1024px, let's set the width to 980px), and then used Media Query to monitor the size change of the browser. When the browser resolution is less than 1024px, Using Media Query's preset style sheet, the width of the page is set to a percentage display so that the structural elements of the page are adjusted according to the size of the browser. Similarly, when the browser's viewable area changes to a certain value (say 650px), the page's structural elements adjust accordingly according to Media Query's preset cascading style sheet.

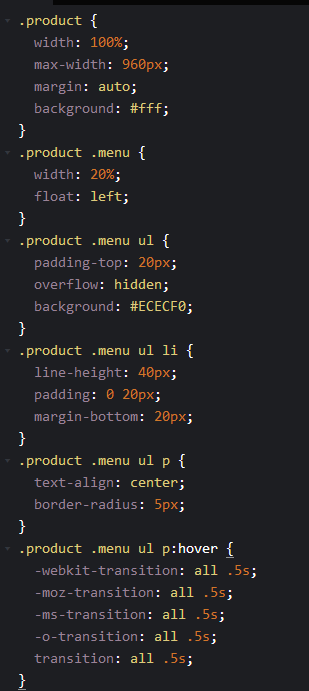
**Task2**

CSS Settings at the top of the site

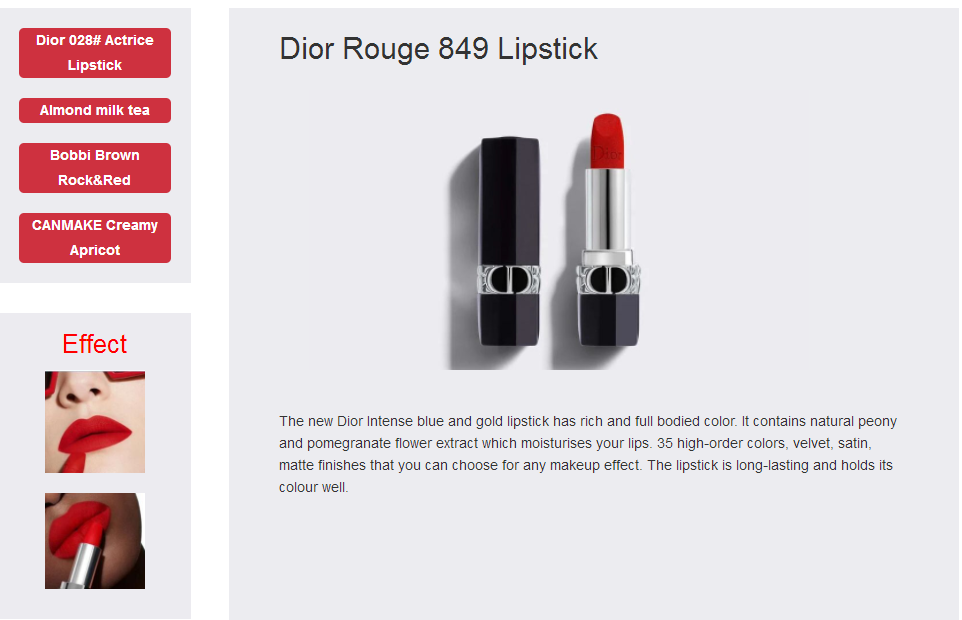


The renderings are as follows: 

Content area CSS Settings



Content body structure and color matching



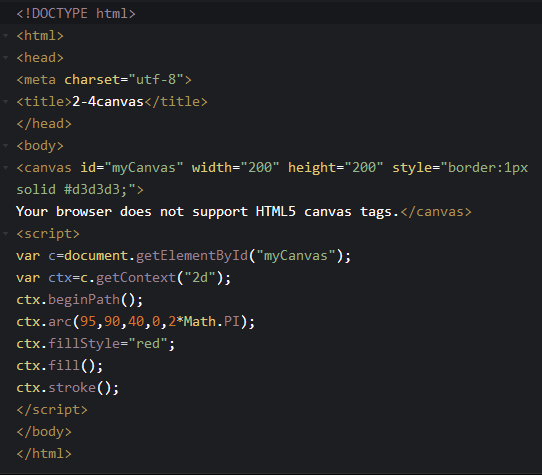
**Task3**

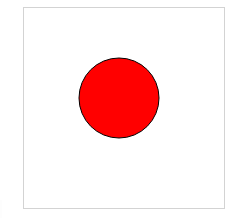
User stories

|  |  |
| --- | --- |
| **Statement** | **Acceptance Criteria** |
| If a user wanted to search for  content or products on the website | After entering keywords in the search bar, the page will be  navigated to the search results. |
| If a user wants to log in to the  website to check cart products | Add a login page, after logging in, you can view your favorite  products and then add to cart. |
| If a user wants to share products as  recommendations | After the login is successful, you can upload the product  picture and fill in the recommended reason. |
| If a user wants to share the website  with any other friends | Add one-click sharing components for social platforms such as  Facebook etc. to the page. |

**Task4**

Canvas Practice





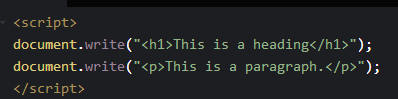
**Reflections:**

This week I learned how to make web pages responsive. And the use of ADVANCED HTML tags. Improved the color and style Settings of the website in the project.

# Week 3 -JavaScript Functions

JavaScript is the most popular scripting language on the Internet. It is used in HTML and the Web, and is widely used on servers, PCS, laptops, tablets and smartphones.

**Task 1**

Write directly to the HTML output stream

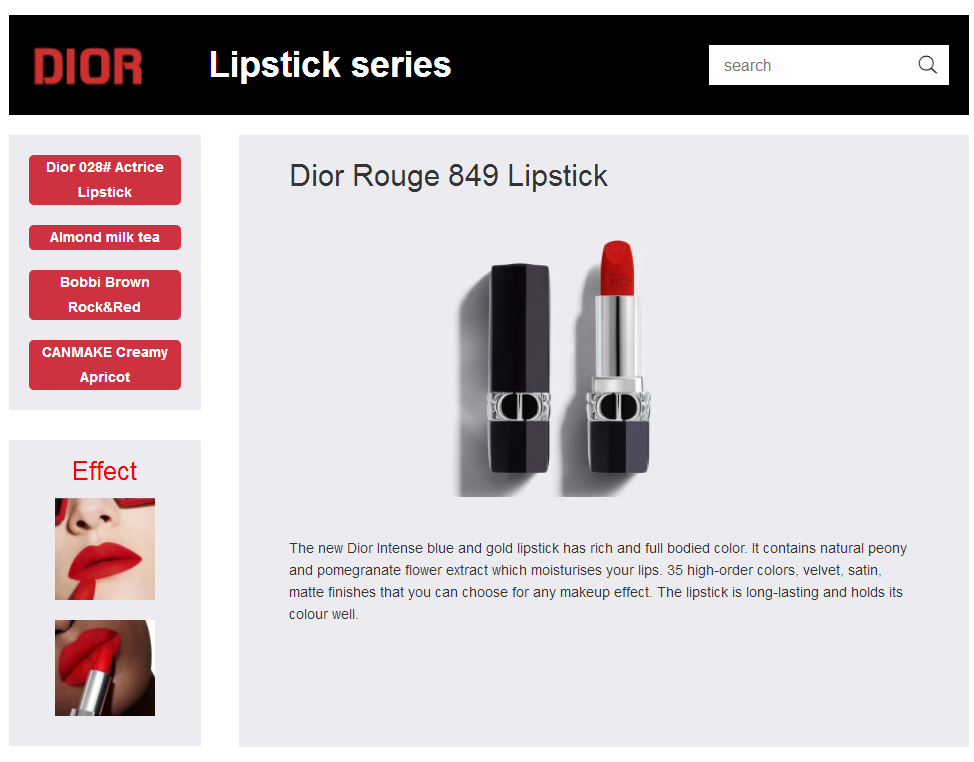


**Lab Exercise Answers**

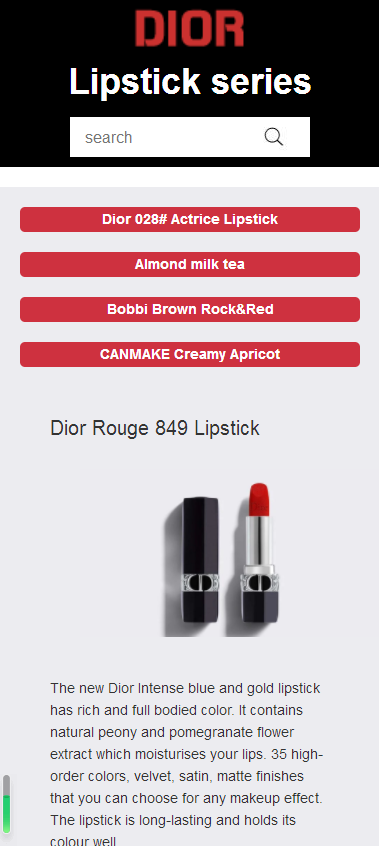
Viewport meta tag

<meta name="viewport" content="width=device-width,initial-scale=1,minimum-scale=1,maximum-scale=1,user-scalable=no">

The display effect of PC terminal and mobile terminal is as follows:

PC station effect :

Phone station effect：



More User stories

|  |  |
| --- | --- |
| **Statement** | **Acceptance Criteria** |
| If a user wants to see more lipsticks in shades and try them out. | Click the lipstick image to switch to the selected color Click the color number lipstick trial image to switch to the  selected color image |
| If a user fills in the order information incorrectly, there will be a prompt to  tell them so | If an option is not filled, the prompt cannot be empty.  A valid phone number needs to be provided. |
| If a user wants to comment on the product after purchasing it | Set up user evaluation column for real evaluation based on your own usage. Before the evaluation, you need to log in to the page or verify the phone to ensure the authenticity of the  evaluation. |

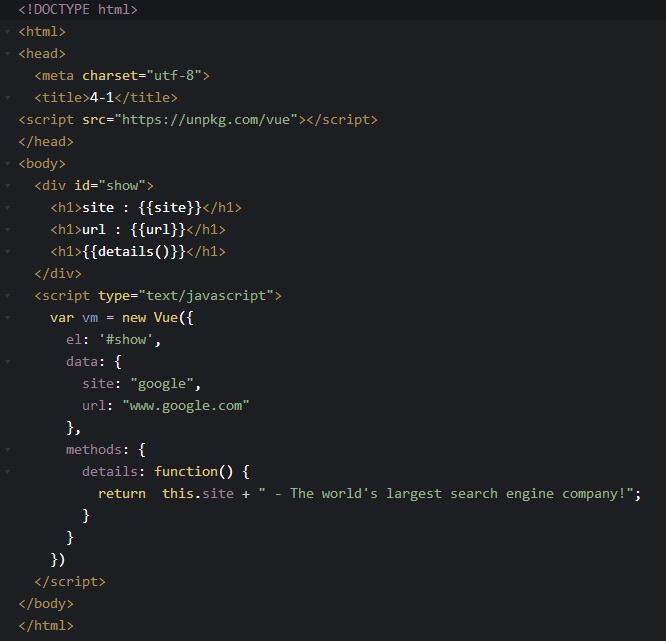
**Reflections:**

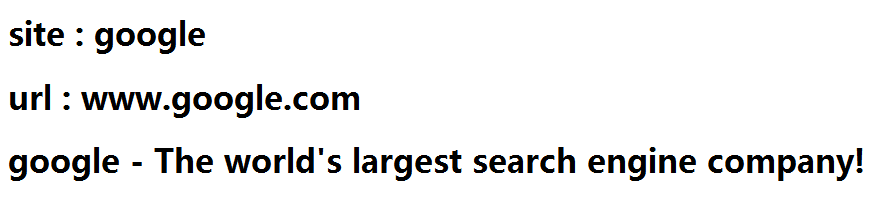
My understanding of responsiveness is that the full page content can be seen on different screen widths. On a smaller screen, the content may be cut, the layout may change, etc. Responsivity is generally implemented using CSS3 media query @media. This week I used responsiveness to perfect my work so that the left and right structure at the top of the content area changes to an up and down structure when the phone visits the page. Buttons on the left are arranged from top to bottom. The Effect on the right side is hidden on the mobile phone due to the length of the screen. The content on the right is displayed under the menu. Achieve adaptive effects.

# Week 4 -JS Frameworks

**Lab Exercise Answers**

Task1

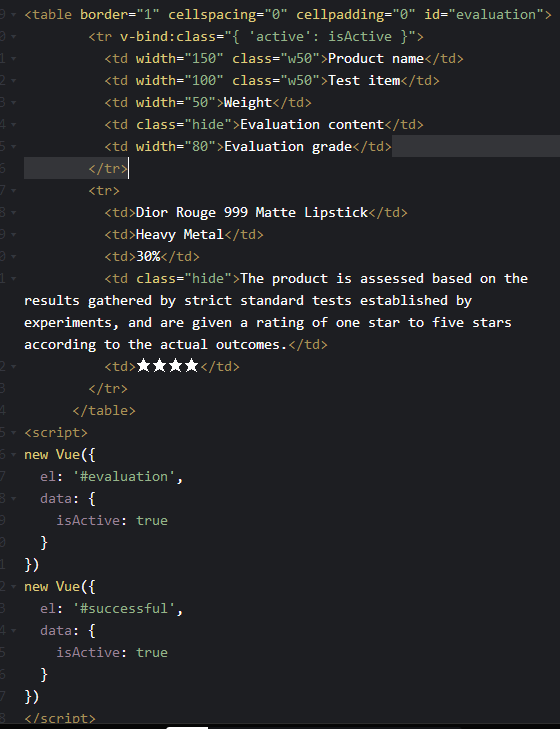
Vue basic display 

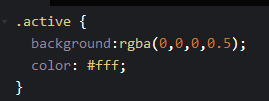


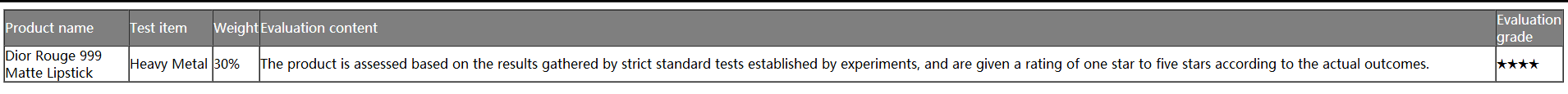
Task 2

|  |  |
| --- | --- |
| statement | Description |
| v-if | The condition judgment uses v-if instruction |
| v-else-if | Else -if block for V-if. Can be chain type multiple use |
| v-show | Display elements according to conditions |
| v-for | Bind data to an array to render a list |

Task3







**Reflections:**

This week I will learn about vUE conditional statements and use V-if instructions to determine conditions. You can use the V-else directive to add an "else" block to v-if as a condition judgment, or you can use v-else-if to perform multiple condition judgments. You can also use the V-show directive to display elements based on conditions. Learn to use v-bind's class to bind styles by judging whether they are displayed.

# Week 5 -Handling User Inputs

Components are one of the most powerful features of vue.js. Components can extend the HTML element, encapsulate reusable code.

**Props** - The scope of a component instance is isolated. You cannot directly reference the parent component's data within the child component's template. Data from the parent component can only be delivered to the child component through prop. Props is the only interface through which the child component accesses the parent component's data. If the child component wants to reference the parent element's data, declare a variable inside prop that can reference the parent element's data. Then render the variable in the template, rendering the parent element's data.

**Custom events** - The parent component uses props to pass data to the child component, but if the child component wants to pass data back, it needs to use custom events. The naming convention of a custom event is different from that of the component registration and props. Because a custom event is an HTML attribute, it is best to use a dash in an HTML template.

**Custom instruction** - In addition to the core directives (V-model and V-show) set by default, Vue also allows registering custom directives. Register a global directive with the vue.directive () function. Add a local directive to the component through the component's directives property.

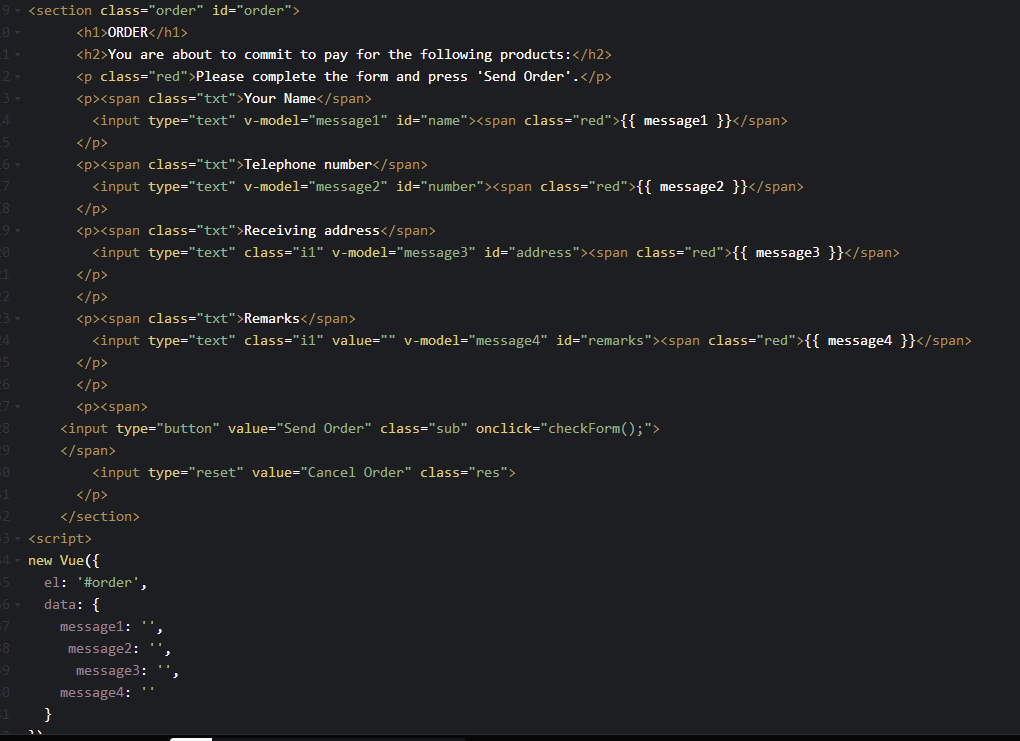
Bind event instances: 

# Week 6 -Composing with components

Vue is a set of progressive frameworks for building user interfaces. Focus only on the view layer and adopt a bottom-up incremental design. Vue goal is through as far as possible simple API data binding and combination of the response of the view components.

**Lab Exercise Answers**

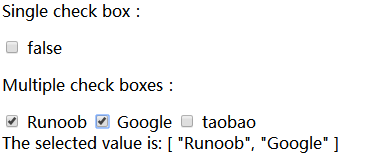
Task1



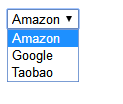


Task2

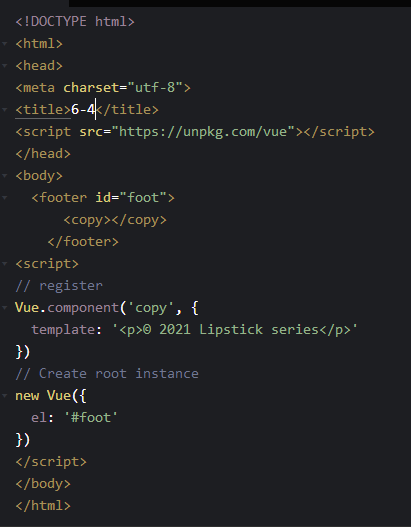


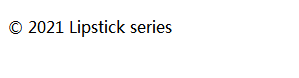


Task3



Task4



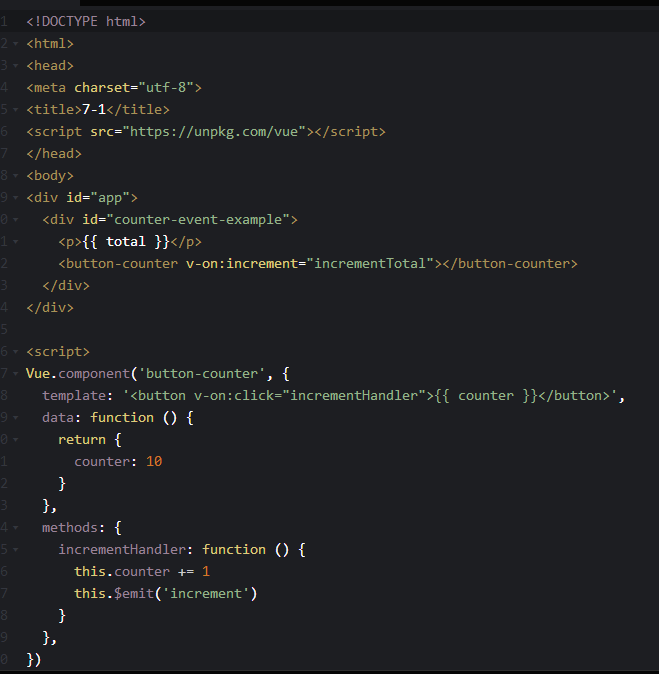


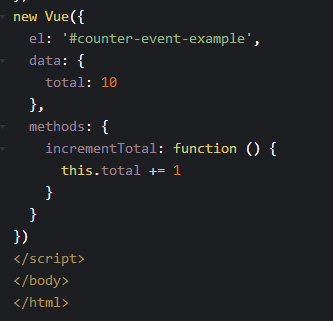
**Reflections:**

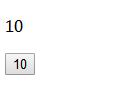
This week I studied vue components in depth, using Vue.com Ponent (tagName, options) to register components, V-for command loop statements, and binding of form elements. Template syntax and bidirectional binding of data.

# Week 7 - Vue Components in-Depth

Task1

Vue custom component instances 





Task2

static props



Results

图片 5

Task3

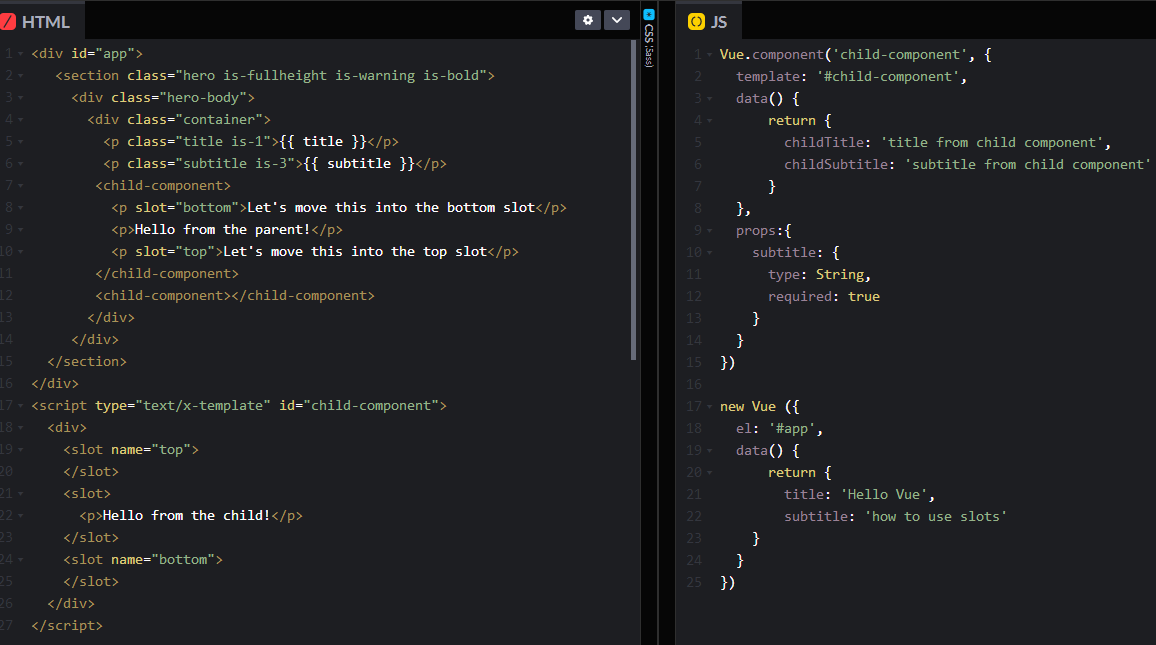
V-model practice



Operation effect Screenshot

图片 7

Task4



result of execution



**Reflections:**

Components are one of Vue's most powerful features. You need to create a component using the Vue. extend method and then register the component using the Vue.component method. All instances can use global components. Local components can also be registered in the instance. Prop is a custom property that a child component uses to accept data passed by the parent component. Componentization is an independent and reusable unit of code organization and using components can greatly improve development efficiency.

<https://github.com/llloralyw/A2-Document-And-Code>