week3

Note:

Task1:

A string can also be referred to as a string for short. In the data structure, a string is a linear table with certain constraints on the composition of data elements, that is, all data elements that make up the linear table are required to be characters, so the string is a finite sequence of characters.

Task2:

This talks about how to use array methods and number methods. We can use these methods to better complete JavaScript.

Task3:

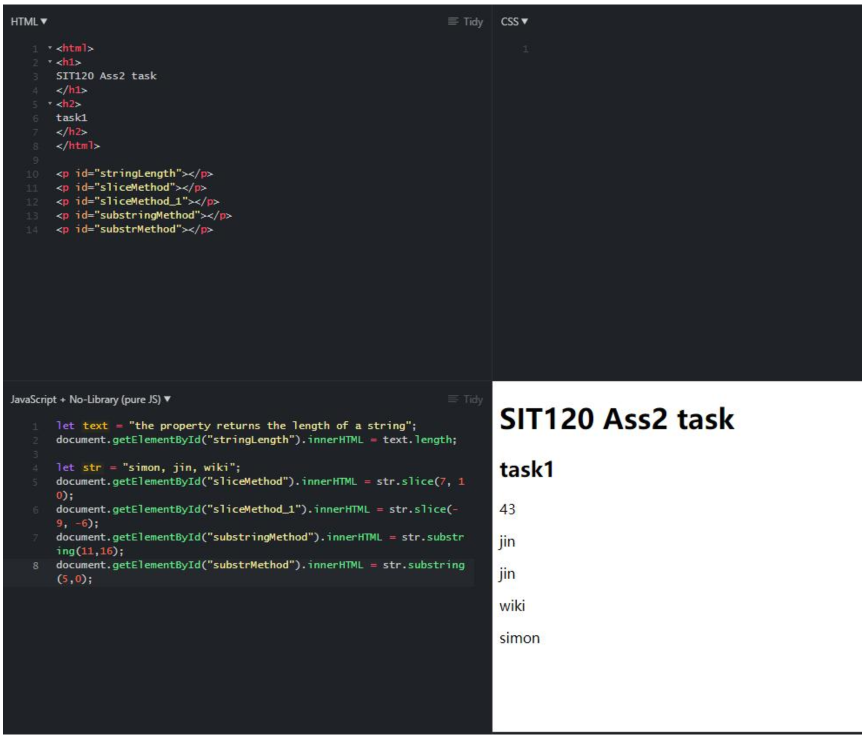
This taught me how to use get and set methods.

Task4:

The explanation of the Vue components and element properties in Task 4 made me learn more and different knowledge, which enabled me to have a deeper understanding of VUE components.

Screenshot:

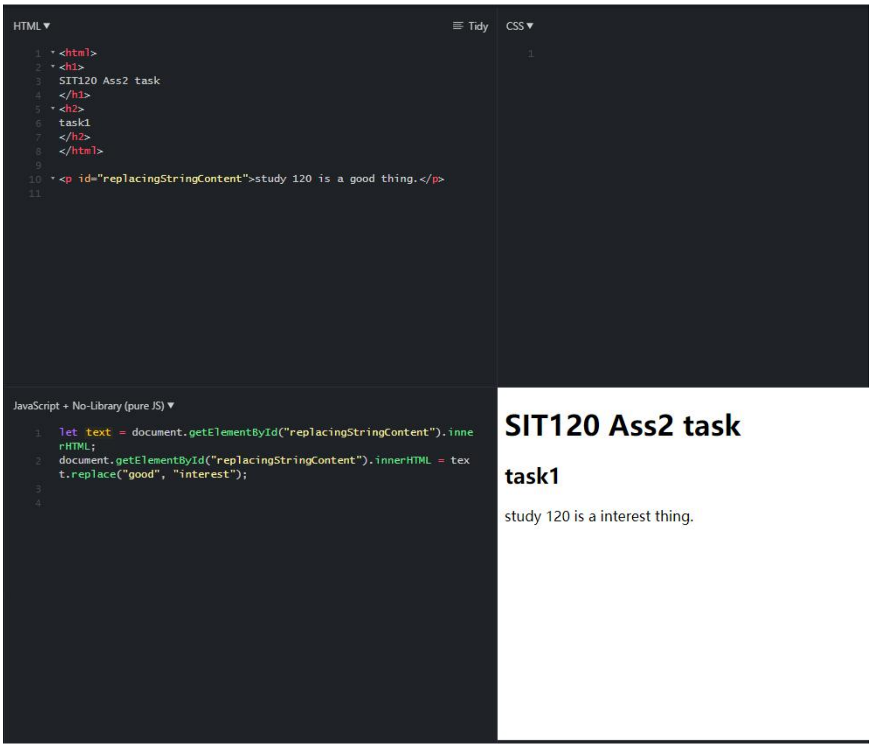
Task1



```html
<html>
<h1>
SIT120 Ass2 task
</h1>
<h2>
task1
</h2>
</html>

<p id="stringLength"></p>
<p id="sliceMethod"></p>
<p id="sliceMethod_1"></p>
<p id="substringMethod"></p>
<p id="substrMethod"></p>
```

```javascript
let text = "the property returns the length of a string";
document.getElementById("stringLength").innerHTML = text.length;

let str = "simon, jin, wiki";
document.getElementById("sliceMethod").innerHTML = str.slice(7, 10);
document.getElementById("sliceMethod_1").innerHTML = str.slice(-9, -6);
document.getElementById("substringMethod").innerHTML = str.substring(11,16);
document.getElementById("substrMethod").innerHTML = str.substring(5,0);
```

```
SIT120 Ass2 task

task1

43

jin

jin

wiki

simon
```



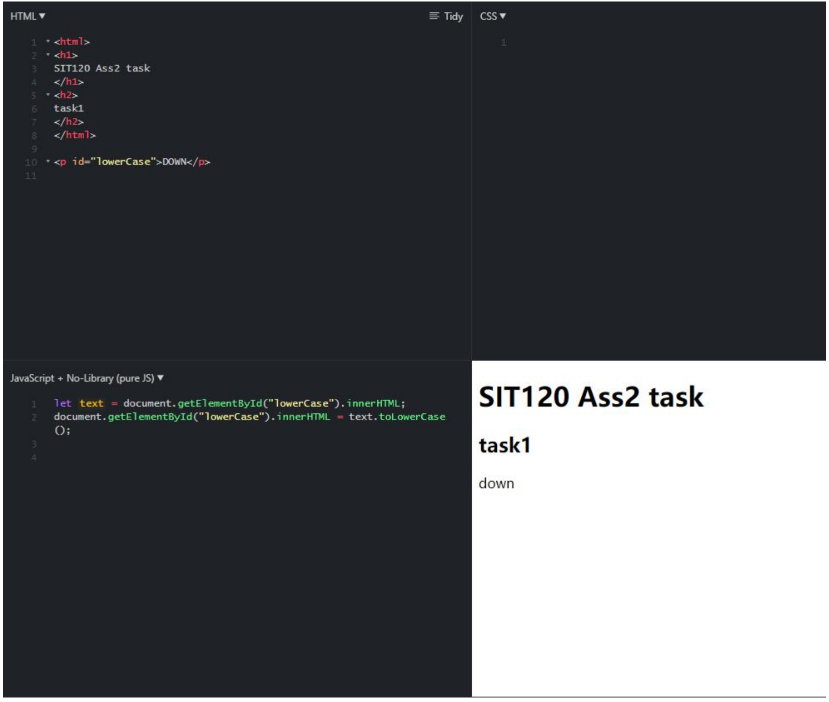
```html
<html>
<h1>
SIT120 Ass2 task
</h1>
<h2>
task1
</h2>
</html>

<p id="replacingStringContent">study 120 is a good thing.</p>
```

```javascript
let text = document.getElementById("replacingStringContent").innerHTML;
document.getElementById("replacingStringContent").innerHTML = text.replace("good", "interest");
```

SIT120 Ass2 task

task1

study 120 is a interest thing.



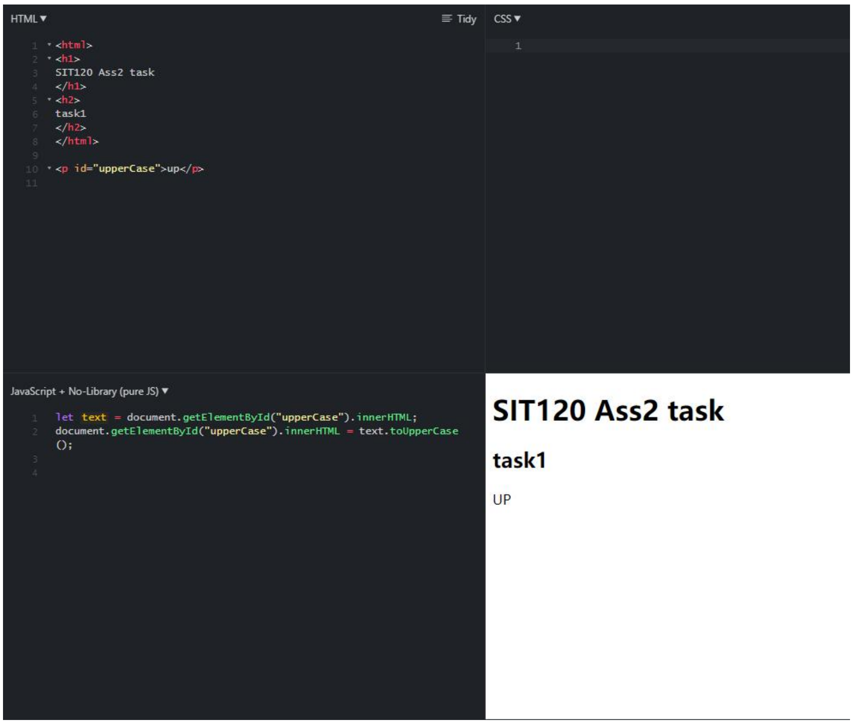
```html
<html>
<h1>
SIT120 Ass2 task
</h1>
<h2>
task1
</h2>
</html>

<p id="lowerCase">DOWN</p>
```

```javascript
let text = document.getElementById("lowerCase").innerHTML;
document.getElementById("lowerCase").innerHTML = text.toLowerCase();
```

SIT120 Ass2 task

task1

down



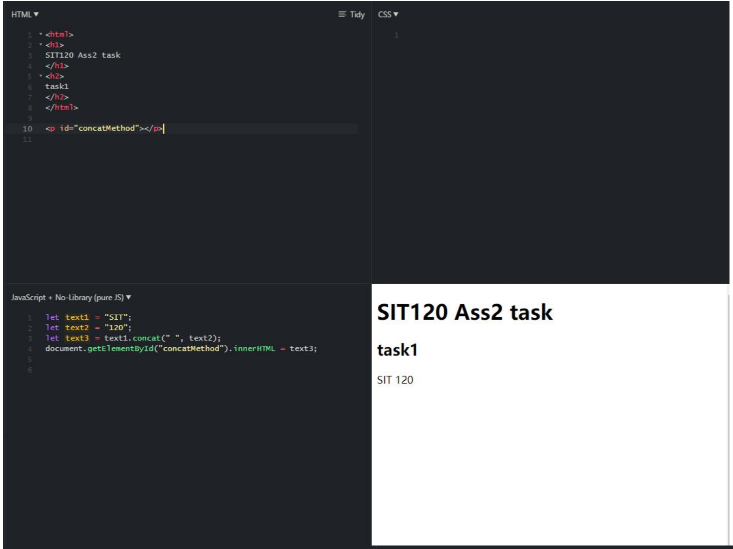
```html
<html>
<h1>
SIT120 Ass2 task
</h1>
<h2>
task1
</h2>
</html>

<p id="upperCase">up</p>
```

```javascript
let text = document.getElementById("upperCase").innerHTML;
document.getElementById("upperCase").innerHTML = text.toUpperCase
();
```

SIT120 Ass2 task

task1

UP



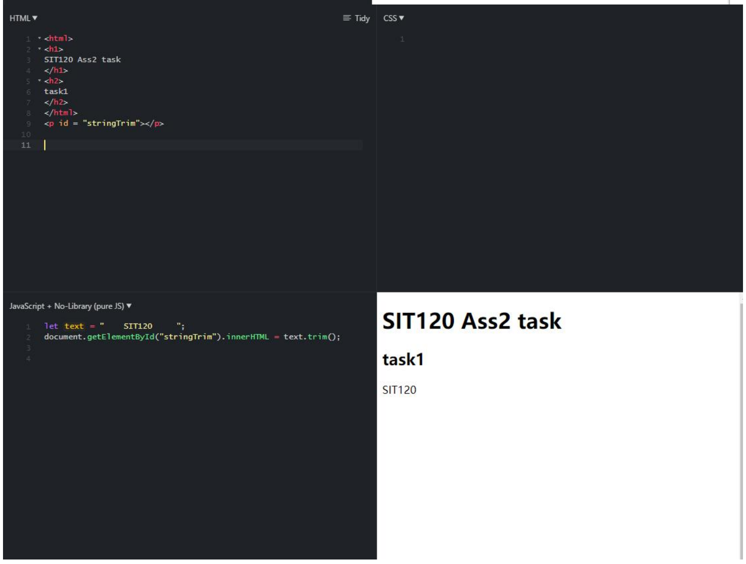
```html
<html>
<h1>
SIT120 Ass2 task
</h1>
<h2>
task1
</h2>
</html>

<p id="concatMethod"></p>
```

```javascript
let text1 = "SIT";
let text2 = "120";
let text3 = text1.concat(" ", text2);
document.getElementById("concatMethod").innerHTML = text3;
```

SIT120 Ass2 task

task1

SIT 120



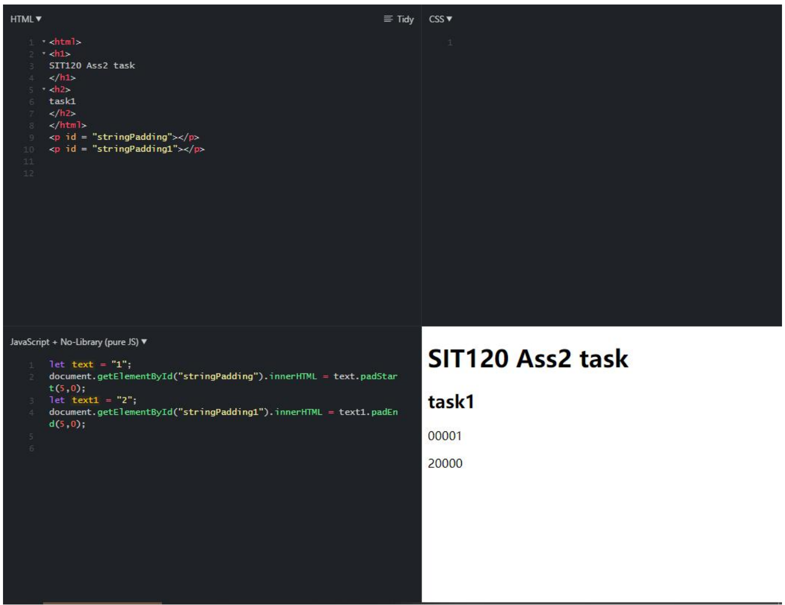
```html
<html>
<h1>
SIT120 Ass2 task
</h1>
<h2>
task1
</h2>
</html>
<p id = "stringTrim"></p>
```

```javascript
let text = "    SIT120     ";
document.getElementById("stringTrim").innerHTML = text.trim();
```

SIT120 Ass2 task

task1

SIT120



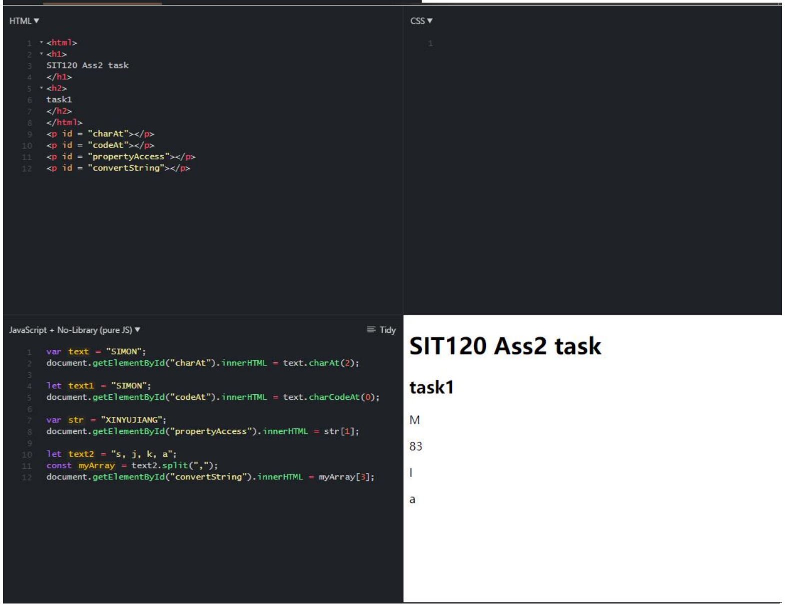
```html
<html>
<h1>
SIT120 Ass2 task
</h1>
<h2>
task1
</h2>
</html>
<p id = "stringPadding"></p>
<p id = "stringPadding1"></p>
```

```javascript
let text = "1";
document.getElementById("stringPadding").innerHTML = text.padStart(5,0);
let text1 = "2";
document.getElementById("stringPadding1").innerHTML = text1.padEnd(5,0);
```

SIT120 Ass2 task

task1

```
00001
20000
```



```html
<html>
<h1>
SIT120 Ass2 task
</h1>
<h2>
task1
</h2>
</html>
<p id = "charAt"></p>
<p id = "codeAt"></p>
<p id = "propertyAccess"></p>
<p id = "convertString"></p>
```

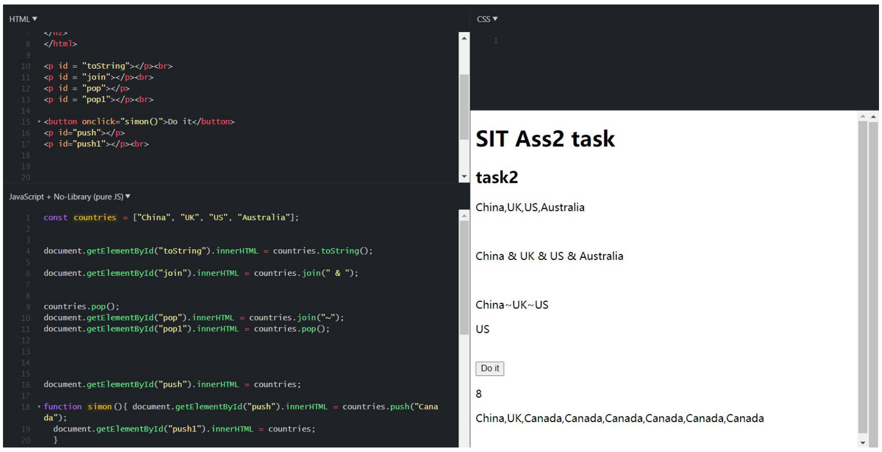
```javascript
var text = "SIMON";
document.getElementById("charAt").innerHTML = text.charAt(2);

let text1 = "SIMON";
document.getElementById("codeAt").innerHTML = text.charCodeAt(0);

var str = "XINYUJIANG";
document.getElementById("propertyAccess").innerHTML = str[1];

let text2 = "s, j, k, a";
const myArray = text2.split(",");
document.getElementById("convertString").innerHTML = myArray[3];
```

SIT120 Ass2 task

task1

```
M
83
I
a
```

Task2



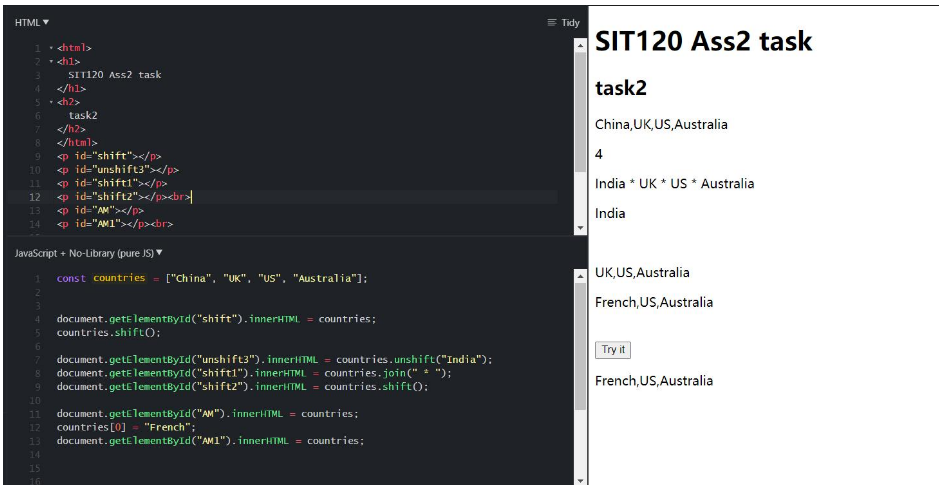
```html
</h2>
</html>

<p id = "toString"></p><br>
<p id = "join"></p><br>
<p id = "pop"></p>
<p id = "pop1"></p><br>

<button onclick="simon()">Do it</button>
<p id="push"></p>
<p id="push1"></p><br>
```

```javascript
const countries = ["China", "UK", "US", "Australia"];


document.getElementById("toString").innerHTML = countries.toString();

document.getElementById("join").innerHTML = countries.join(" & ");


countries.pop();
document.getElementById("pop").innerHTML = countries.join("~");
document.getElementById("pop1").innerHTML = countries.pop();




document.getElementById("push").innerHTML = countries;

function simon(){ document.getElementById("push").innerHTML = countries.push("Canada");
  document.getElementById("push1").innerHTML = countries;
  }
```

SIT Ass2 task

task2

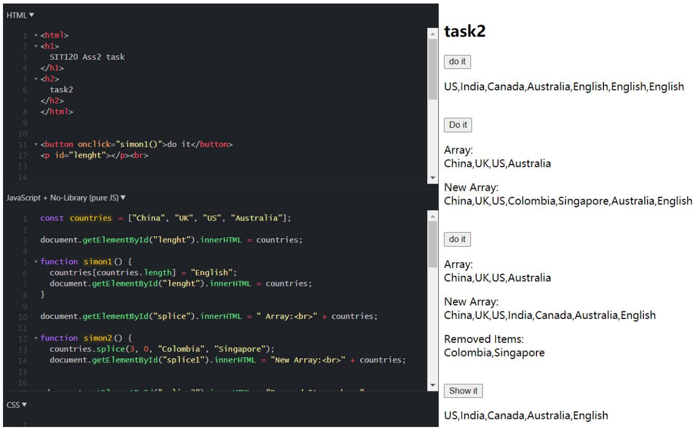
```
China,UK,US,Australia

China & UK & US & Australia

China~UK~US
US

Do it
8
China,UK,Canada,Canada,Canada,Canada,Canada,Canada
```



```html
<html>
<h1>
  SIT120 Ass2 task
</h1>
<h2>
  task2
</h2>
</html>
<p id="shift"></p>
<p id="unshift3"></p>
<p id="shift1"></p>
<p id="shift2"></p><br>
<p id="AM"></p>
<p id="AM1"></p><br>
```

```javascript
const countries = ["China", "UK", "US", "Australia"];


document.getElementById("shift").innerHTML = countries;
countries.shift();

document.getElementById("unshift3").innerHTML = countries.unshift("India");
document.getElementById("shift1").innerHTML = countries.join(" * ");
document.getElementById("shift2").innerHTML = countries.shift();

document.getElementById("AM").innerHTML = countries;
countries[0] = "French";
document.getElementById("AM1").innerHTML = countries;
```

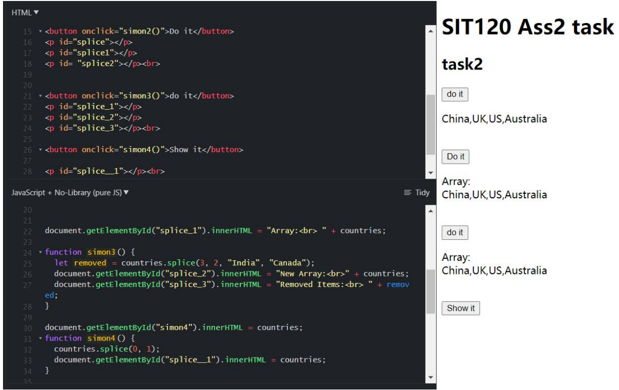
```
SIT120 Ass2 task

task2

China,UK,US,Australia

4

India * UK * US * Australia

India


UK,US,Australia

French,US,Australia

[Try it]

French,US,Australia
```



```html
<html>
<h1>
  SIT120 Ass2 task
</h1>
<h2>
  task2
</h2>
</html>


<button onclick="simon1()">do it</button>
<p id="lenght"></p><br>
```

```javascript
const countries = ["China", "UK", "US", "Australia"];

document.getElementById("lenght").innerHTML = countries;

function simon1() {
  countries[countries.length] = "English";
  document.getElementById("lenght").innerHTML = countries;
}

document.getElementById("splice").innerHTML = " Array:<br>" + countries;

function simon2() {
  countries.splice(3, 0, "Colombia", "Singapore");
  document.getElementById("splice1").innerHTML = "New Array:<br>" + countries;
```

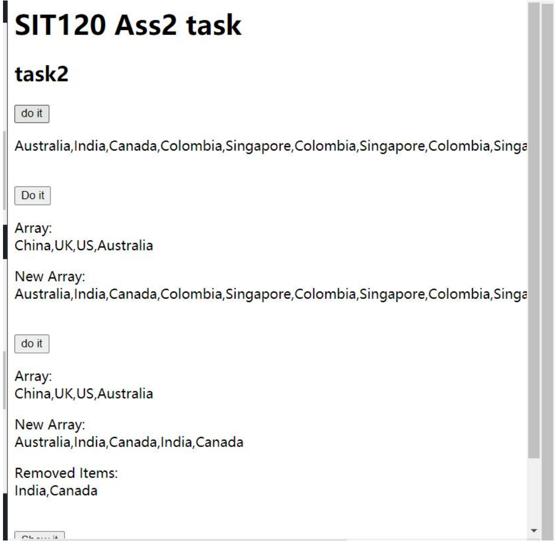
```
task2

[do it]

US,India,Canada,Australia,English,English,English


[Do it]

Array:
China,UK,US,Australia

New Array:
China,UK,US,Colombia,Singapore,Australia,English


[do it]

Array:
China,UK,US,Australia

New Array:
China,UK,US,India,Canada,Australia,English

Removed Items:
Colombia,Singapore

[Show it]

US,India,Canada,Australia,English
```



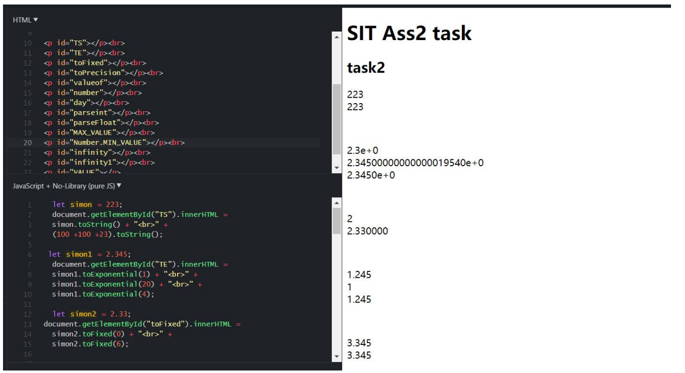
```html
<button onclick="simon2()">Do it</button>
<p id="splice"></p>
<p id="splice1"></p>
<p id= "splice2"></p><br>


<button onclick="simon3()">do it</button>
<p id="splice_1"></p>
<p id="splice_2"></p>
<p id="splice_3"></p><br>

<button onclick="simon4()">Show it</button>

<p id="splice__1"></p><br>
```

```javascript


document.getElementById("splice_1").innerHTML = "Array:<br> " + countries;

function simon3() {
  let removed = countries.splice(3, 2, "India", "Canada");
  document.getElementById("splice_2").innerHTML = "New Array:<br>" + countries;
  document.getElementById("splice_3").innerHTML = "Removed Items:<br> " + remov
ed;
}

document.getElementById("simon4").innerHTML = countries;
function simon4() {
  countries.splice(0, 1);
  document.getElementById("splice__1").innerHTML = countries;
}
```

```
SIT120 Ass2 task

task2

[do it]

China,UK,US,Australia


[Do it]

Array:
China,UK,US,Australia


[do it]

Array:
China,UK,US,Australia


[Show it]
```



```
SIT120 Ass2 task

task2

[do it]

Australia,India,Canada,Colombia,Singapore,Colombia,Singapore,Colombia,Singa


[Do it]

Array:
China,UK,US,Australia

New Array:
Australia,India,Canada,Colombia,Singapore,Colombia,Singapore,Colombia,Singa


[do it]

Array:
China,UK,US,Australia

New Array:
Australia,India,Canada,India,Canada

Removed Items:
India,Canada
```



HTML ▼

```html
<p id="TS"></p><br>
<p id="TE"></p><br>
<p id="toFixed"></p><br>
<p id="toPrecision"></p><br>
<p id="valueof"></p><br>
<p id="number"></p><br>
<p id="day"></p><br>
<p id="parseint"></p><br>
<p id="parseFloat"></p><br>
<p id="MAX_VALUE"></p><br>
<p id="Number.MIN_VALUE"></p><br>
<p id="infinity"></p><br>
<p id="infinity1"></p><br>
<p id="VALUE"></p>
```

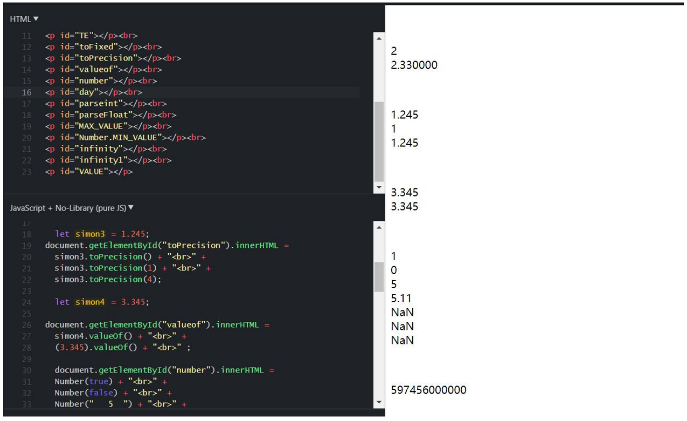
JavaScript + No-Library (pure JS) ▼

```javascript
  let simon = 223;
  document.getElementById("TS").innerHTML =
  simon.toString() + "<br>" +
  (100 +100 +23).toString();

 let simon1 = 2.345;
  document.getElementById("TE").innerHTML =
  simon1.toExponential(1) + "<br>" +
  simon1.toExponential(20) + "<br>" +
  simon1.toExponential(4);

  let simon2 = 2.33;
document.getElementById("toFixed").innerHTML =
  simon2.toFixed(0) + "<br>" +
  simon2.toFixed(6);
```

SIT Ass2 task

task2

```
223
223


2.3e+0
2.34500000000000019540e+0
2.3450e+0


2
2.330000


1.245
1
1.245


3.345
3.345
```



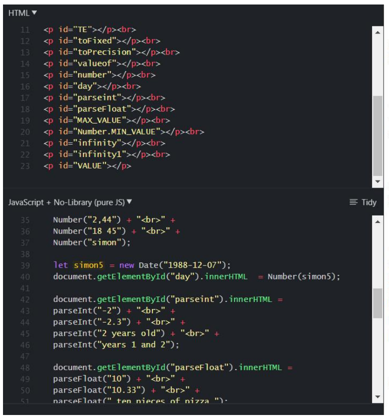
HTML ▼

```html
<p id="TE"></p><br>
<p id="toFixed"></p><br>
<p id="toPrecision"></p><br>
<p id="valueof"></p><br>
<p id="number"></p><br>
<p id="day"></p><br>
<p id="parseint"></p><br>
<p id="parseFloat"></p><br>
<p id="MAX_VALUE"></p><br>
<p id="Number.MIN_VALUE"></p><br>
<p id="infinity"></p><br>
<p id="infinity1"></p><br>
<p id="VALUE"></p>
```

JavaScript + No-Library (pure JS) ▼

```javascript
  let simon3 = 1.245;
document.getElementById("toPrecision").innerHTML =
  simon3.toPrecision() + "<br>" +
  simon3.toPrecision(1) + "<br>" +
  simon3.toPrecision(4);

  let simon4 = 3.345;

document.getElementById("valueof").innerHTML =
  simon4.valueOf() + "<br>" +
  (3.345).valueOf() + "<br>" ;

  document.getElementById("number").innerHTML =
  Number(true) + "<br>" +
  Number(false) + "<br>" +
  Number("   5  ") + "<br>" +
```

```
2
2.330000


1.245
1
1.245


3.345
3.345


1
0
5
5.11
NaN
NaN
NaN


597456000000
```



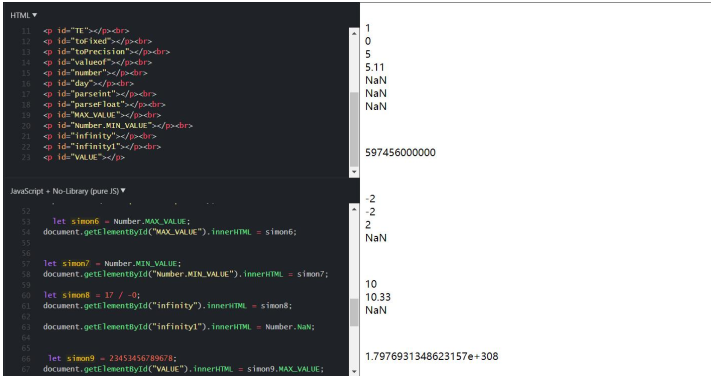
HTML ▼

```html
<p id="TE"></p><br>
<p id="toFixed"></p><br>
<p id="toPrecision"></p><br>
<p id="valueof"></p><br>
<p id="number"></p><br>
<p id="day"></p><br>
<p id="parseint"></p><br>
<p id="parseFloat"></p><br>
<p id="MAX_VALUE"></p><br>
<p id="Number.MIN_VALUE"></p><br>
<p id="infinity"></p><br>
<p id="infinity1"></p><br>
<p id="VALUE"></p>
```

JavaScript + No-Library (pure JS) ▼ ≡ Tidy

```javascript
  Number("2,44") + "<br>" +
  Number("18 45") + "<br>" +
  Number("simon");

  let simon5 = new Date("1988-12-07");
  document.getElementById("day").innerHTML  = Number(simon5);

  document.getElementById("parseint").innerHTML =
  parseInt("-2") + "<br>" +
  parseInt("-2.3") + "<br>" +
  parseInt("2 years old") + "<br>" +
  parseInt("years 1 and 2");

  document.getElementById("parseFloat").innerHTML =
  parseFloat("10") + "<br>" +
  parseFloat("10.33") + "<br>" +
  parseFloat(" ten pieces of pizza ");
```

```
2
2.330000


1.245
1
1.245


3.345
3.345


1
0
5
5.11
NaN
NaN
NaN


597456000000
```



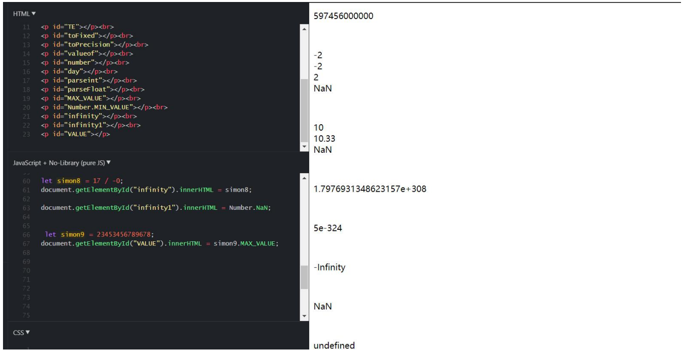
HTML ▼

```html
<p id="TE"></p><br>
<p id="toFixed"></p><br>
<p id="toPrecision"></p><br>
<p id="valueof"></p><br>
<p id="number"></p><br>
<p id="day"></p><br>
<p id="parseint"></p><br>
<p id="parseFloat"></p><br>
<p id="MAX_VALUE"></p><br>
<p id="Number.MIN_VALUE"></p><br>
<p id="infinity"></p><br>
<p id="infinity1"></p><br>
<p id="VALUE"></p>
```

JavaScript + No-Library (pure JS) ▼

```javascript

  let simon6 = Number.MAX_VALUE;
document.getElementById("MAX_VALUE").innerHTML = simon6;


let simon7 = Number.MIN_VALUE;
document.getElementById("Number.MIN_VALUE").innerHTML = simon7;

let simon8 = 17 / -0;
document.getElementById("infinity").innerHTML = simon8;

document.getElementById("infinity1").innerHTML = Number.NaN;


 let simon9 = 23453456789678;
document.getElementById("VALUE").innerHTML = simon9.MAX_VALUE;
```

```
1
0
5
5.11
NaN
NaN
NaN


597456000000


-2
-2
2
NaN


10
10.33
NaN


1.7976931348623157e+308
```



HTML ▼

```html
<p id="TE"></p><br>
<p id="toFixed"></p><br>
<p id="toPrecision"></p><br>
<p id="valueof"></p><br>
<p id="number"></p><br>
<p id="day"></p><br>
<p id="parseint"></p><br>
<p id="parseFloat"></p><br>
<p id="MAX_VALUE"></p><br>
<p id="Number.MIN_VALUE"></p><br>
<p id="infinity"></p><br>
<p id="infinity1"></p><br>
<p id="VALUE"></p>
```

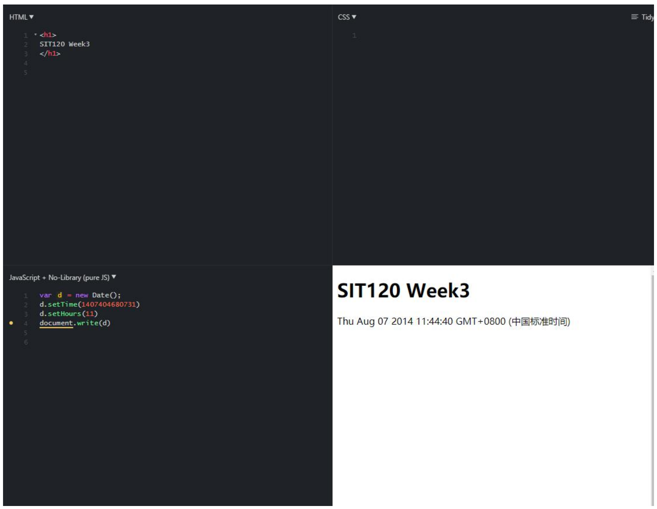
JavaScript + No-Library (pure JS) ▼

```javascript
let simon8 = 17 / -0;
document.getElementById("infinity").innerHTML = simon8;

document.getElementById("infinity1").innerHTML = Number.NaN;


 let simon9 = 23453456789678;
document.getElementById("VALUE").innerHTML = simon9.MAX_VALUE;
```

CSS ▼

```
597456000000


-2
-2
2
NaN


10
10.33
NaN


1.7976931348623157e+308


5e-324


-Infinity


NaN


undefined
```

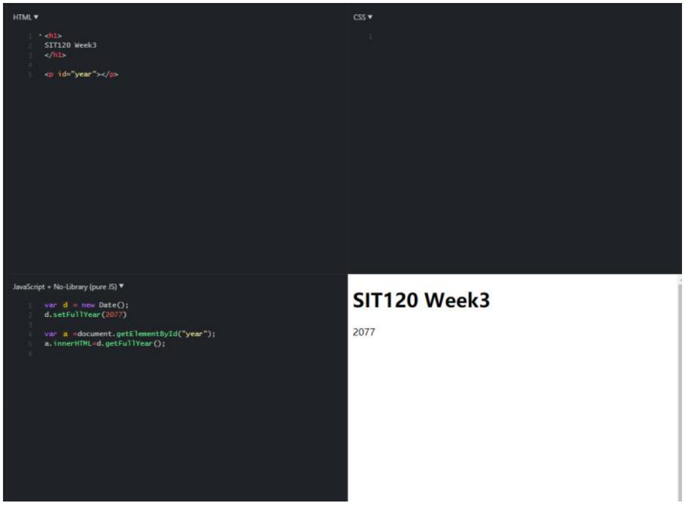
Task3



```html
<h1>
SIT120 Week3
</h1>
```

```javascript
var d = new Date();
d.setTime(1407404680731)
d.setHours(11)
document.write(d)
```

```
SIT120 Week3

Thu Aug 07 2014 11:44:40 GMT+0800 (中国标准时间)
```



```html
<h1>
SIT120 Week3
</h1>

<p id="year"></p>
```

```javascript
var d = new Date();
d.setFullYear(2077)

var a =document.getElementById("year");
a.innerHTML=d.getFullYear();
```

```
SIT120 Week3

2077
```

Task4

Computer attributes :

It is generally used to describe that the value of one attribute depends on the value of another attribute. When we use an interpolation expression to associate a calculated attribute to a page element, the calculated attribute will automatically update the DOM element when the value of the dependent attribute changes.Look at the listener method for monitoring changes to certain data. The difference is that the calculated attributes are only data operations performed after the dependent data changes, while the clock focuses more on a series of logical business operations performed after a monitoring data change.In general: the attribute attribute feature is to save the result and recalculate when the attribute depends on changes. The difference is that the listener focuses on monitoring the real-time changes of certain data and executes some specific business logic of calls, which is not limited.To return the data.

Class and Style Bindings:

Like containers, Classes contain many object behaviors of elements, names, properties, methods, and events.
A class encapsulates the properties and methods to operate on its own members.
Class and Style Binding
In general, Class and Style Binding can help users bind a list of data element classes and their inline styles.

List Rendering:

The most important element of this attribute is the v-for statement, and the V-for element is used to display a list.
But executing the v-for statement requires a special syntax "item in items", items refer to the source data array, and item refers to the alias of the array element to be iterated. List rendering puts the data in the specified location, simplifying the code load and improving the display of the loop.

Event Handling:

In Workflow Planning, I learn how to use situation analysis, process analysis, linear procedures, and learn how to use process analysis and process modifications.

Form Input Bindings:

For bidirectional data binding, we can use the v-model command to bind the form elements <input>, <textarea> and <select>. Bidirectional binding can ensure that each form control has a corresponding record variable in memory.This match is bidirectional, no matter which part changes, the other party will update the corresponding value.

Components Basics:

There are many useful components in this building such as reusable components, organizing components, props guide, Single Root Element, Listening to Kids Events Events, dynamic components, these parts make our code quicker and easier to control.

Component Registration:

Component registration is mainly divided into Global Registration and Local Registration. In general, global registration is undesirable as all components can still be included in your final build even you stopped them. This will allow users to expand the number of irrelevant
JavaScript downloads which will waste users' time and energy. Local registration means to register and import in the Vue file you want to use. Vue. Component will be used when registering global components.

Props:

Props (which includes camelCase vs kebab case), static props, dynamic props, props verification, one-way data flow and the non-props attributes. Static props and dynamic props which dynamically ties up the data of the parent component to the props of the child component template by using v-bind, both can progress numbers, Boolean values, arrays, objects and object attributes. Props verification means when substantiating the data specification of the incoming props parameter, Vue will emerge a warning to tell the user if it does not meet the data specification. All props form a one-way downward binding betwixt child attributes and parent attributes. When the parent property is updated, it will flow down to the child property, but not vice versa. The purpose is to prevent the child component from accidentally changing the state of the parent component. Non props attribute is passed to components, but no corresponding props are defined.

Custom Events：

This module contains many useful modules such as event names.

Slots：

If we want to display the inserted new label in the subcomponent, we can use slots.

Dynamic & Async Components：

Inactive component instances can be saved by him. We can use Async Components. It can save us a lot of work.

Handling Edge Cases

It include element and component access.   Accessing the Root Instance is very convenient.
In general: the small components of this module can help users to manipulate the code more easily.

Reflection:

I learned a lot of JS methods by doing these tasks this week. For example String_Methods, Number_Methods, Array_Methods, Gett_Methods, and Set_Methods. I also learned Number_Methods and Array_Methods of JS. For Task 3, I learned how to use the set/get method to make my code more regular and organized. Task 4 is very helpful for the improvement of my Assignment 1. Let me have a detailed and comprehensive understanding of Vue. It makes my web page more complete and interesting. It took me about 5 hours. For the task of next week, I still need to preview in advance and spend more effort to complete it.