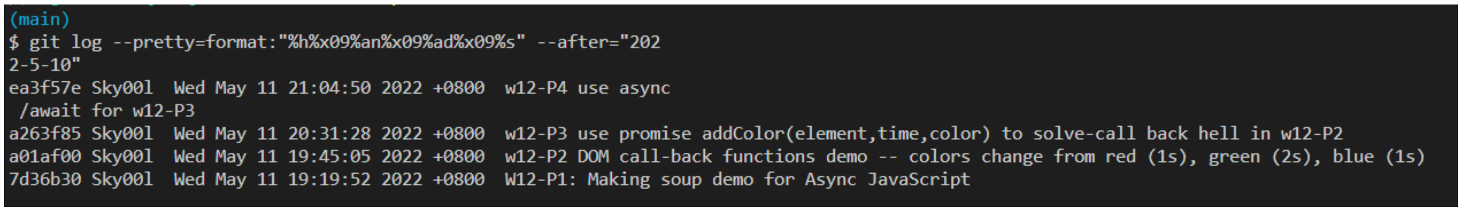
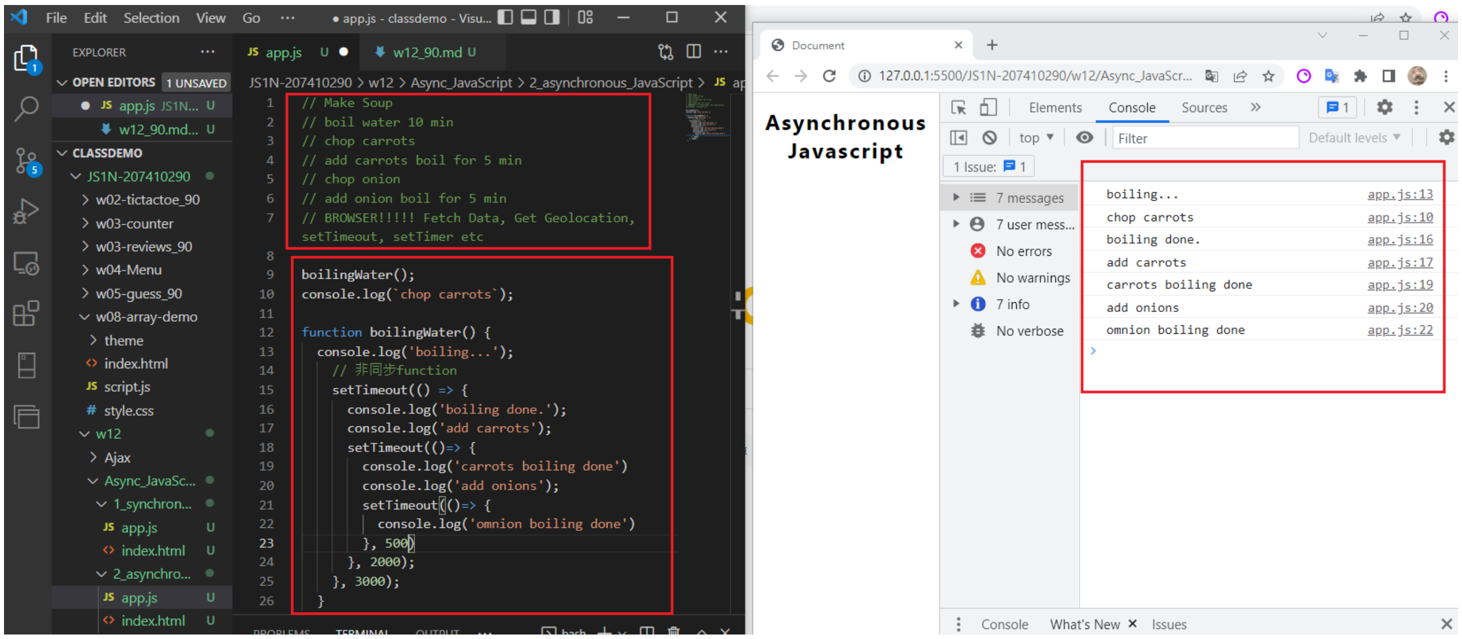
LAST log on "2022-05-11"



```
(main)
$ git log --pretty=format:"%h%x09%an%x09%ad%x09%s" --after="202
2-5-10"
ea3f57e Sky001  Wed May 11 21:04:50 2022 +0800  w12-P4 use async
 /await for w12-P3
a263f85 Sky001  Wed May 11 20:31:28 2022 +0800  w12-P3 use promise addColor(element,time,color) to solve-call back hell in w12-P2
a01af00 Sky001  Wed May 11 19:45:05 2022 +0800  w12-P2 DOM call-back functions demo -- colors change from red (1s), green (2s), blue (1s)
7d36b30 Sky001  Wed May 11 19:19:52 2022 +0800  W12-P1: Making soup demo for Async JavaScript
```

```
$ git log --pretty=format:"%h%x09%an%x09%ad%x09%s" --after="202
2-5-10"
ea3f57e Sky001  Wed May 11 21:04:50 2022 +0800  w12-P4 use async /await for w12-P3
a263f85 Sky001  Wed May 11 20:31:28 2022 +0800  w12-P3 use promise addColor(element,time,color) to solve-call bac
a01af00 Sky001  Wed May 11 19:45:05 2022 +0800  w12-P2 DOM call-back functions demo -- colors change from red (1s
7d36b30 Sky001  Wed May 11 19:19:52 2022 +0800  W12-P1: Making soup demo for Async JavaScript
```

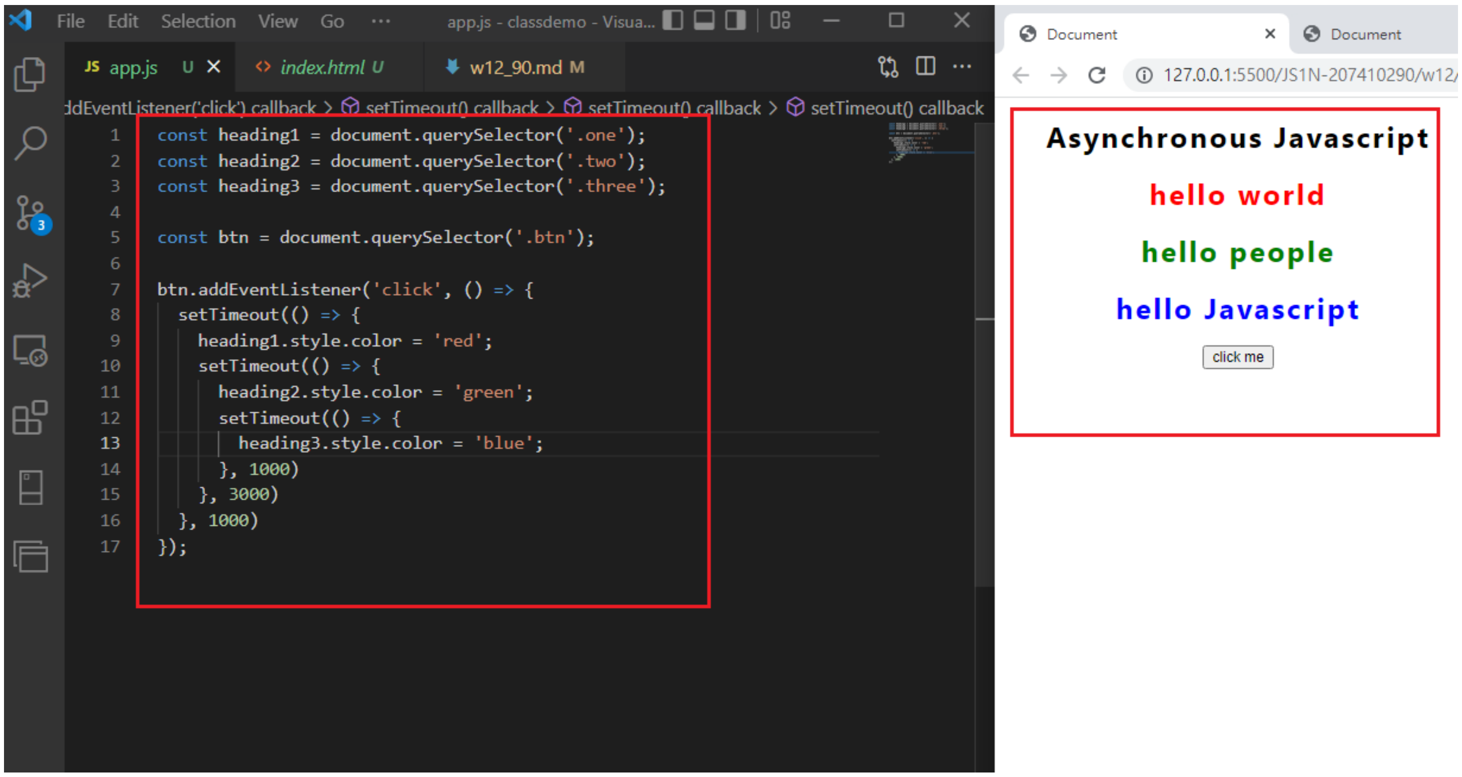
W12-P1: Making soup demo for Async JavaScript with log info



```
1  // Make Soup
2  // boil water 10 min
3  // chop carrots
4  // add carrots boil for 5 min
5  // chop onion
6  // add onion boil for 5 min
7  // BROWSER!!!!! Fetch Data, Get Geolocation,
   setTimeout, setTimer etc
8
9  boilingWater();
10 console.log(`chop carrots`);
11
12 function boilingWater() {
13   console.log('boiling...');
14     // 非同步function
15     setTimeout(() => {
16       console.log('boiling done.');
17       console.log('add carrots');
18       setTimeout(()=> {
19         console.log('carrots boiling done')
20         console.log('add onions');
21         setTimeout(()=> {
22           console.log('omnion boiling done')
23         }, 500)
24       }, 2000);
25     }, 3000);
26   }
```

```
boiling...                app.js:13
chop carrots              app.js:10
boiling done.             app.js:16
add carrots               app.js:17
carrots boiling done      app.js:19
add onions                app.js:20
omnion boiling done       app.js:22
>
```

w12-P2 DOM call-back functions demo -- colors change from red (1s), green (2s), blue (1s)



```javascript
const heading1 = document.querySelector('.one');
const heading2 = document.querySelector('.two');
const heading3 = document.querySelector('.three');

const btn = document.querySelector('.btn');

btn.addEventListener('click', () => {
  setTimeout(() => {
    heading1.style.color = 'red';
    setTimeout(() => {
      heading2.style.color = 'green';
      setTimeout(() => {
        heading3.style.color = 'blue';
      }, 1000)
    }, 3000)
  }, 1000)
});
```

Asynchronous Javascript

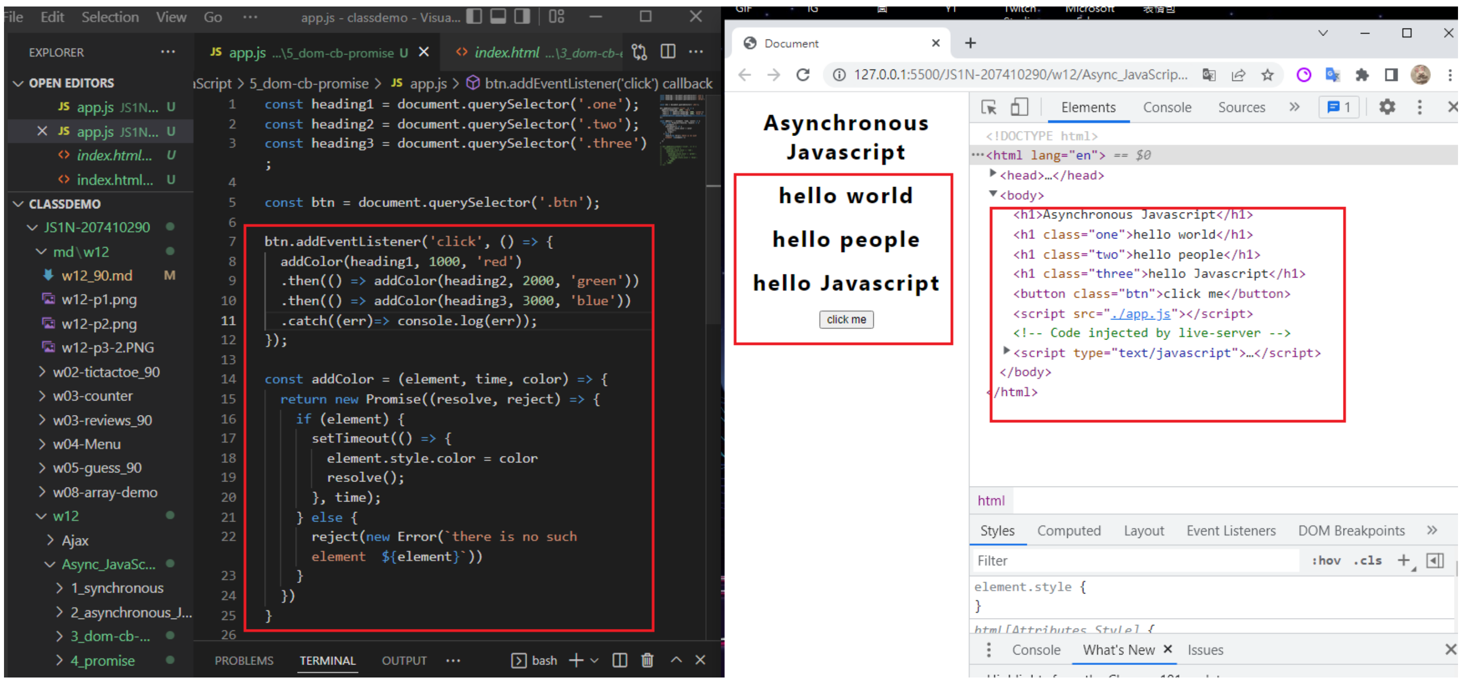
hello world

hello people

hello Javascript

click me

w12-P3 use promise addColor(element,time,color) to solve-call back hell in w12-P2



```javascript
const heading1 = document.querySelector('.one');
const heading2 = document.querySelector('.two');
const heading3 = document.querySelector('.three')
;

const btn = document.querySelector('.btn');

btn.addEventListener('click', () => {
  addColor(heading1, 1000, 'red')
  .then(() => addColor(heading2, 2000, 'green'))
  .then(() => addColor(heading3, 3000, 'blue'))
  .catch((err)=> console.log(err));
});

const addColor = (element, time, color) => {
  return new Promise((resolve, reject) => {
    if (element) {
      setTimeout(() => {
        element.style.color = color
        resolve();
      }, time);
    } else {
      reject(new Error(`there is no such
      element  ${element}`))
    }
  })
}
```

Asynchronous Javascript

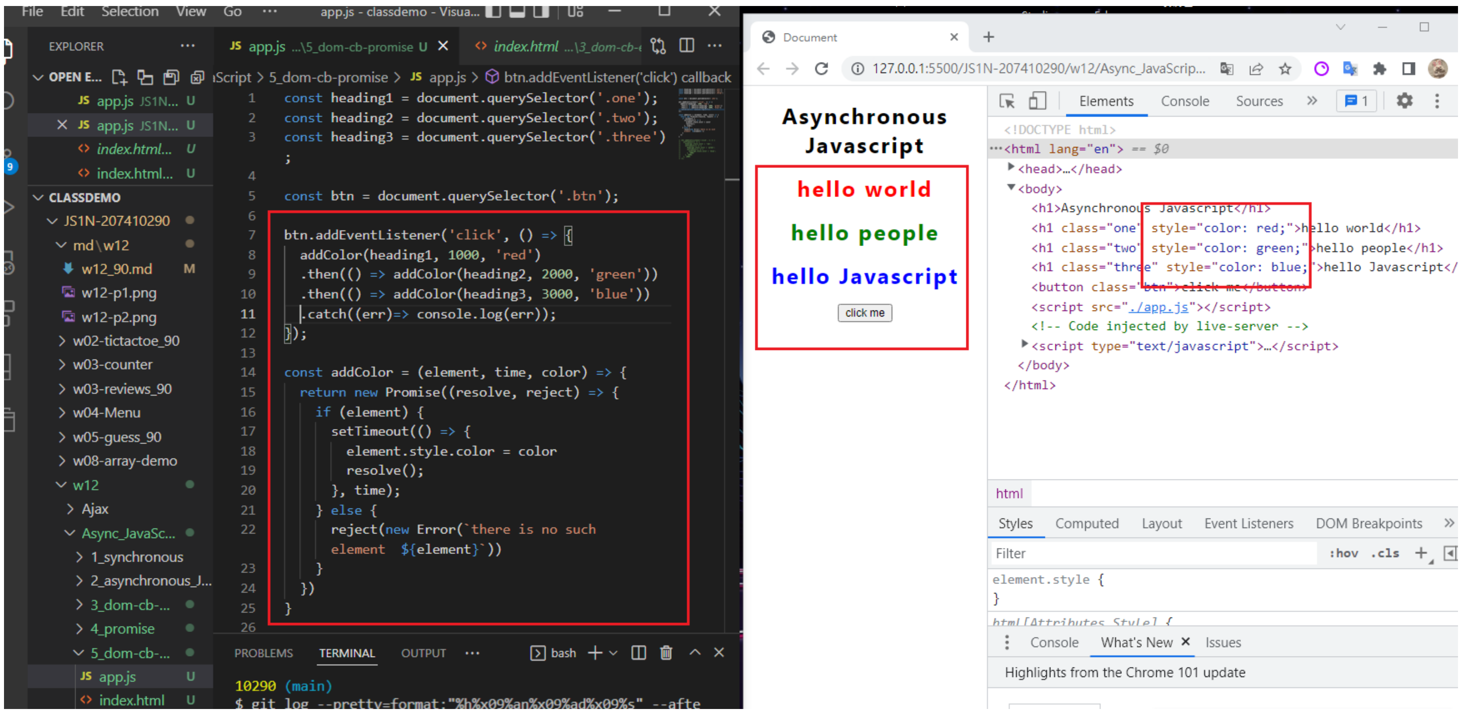
hello world

hello people

hello Javascript

click me

```html
<!DOCTYPE html>
<html lang="en"> == $0
  <head>…</head>
  <body>
    <h1>Asynchronous Javascript</h1>
    <h1 class="one">hello world</h1>
    <h1 class="two">hello people</h1>
    <h1 class="three">hello Javascript</h1>
    <button class="btn">click me</button>
    <script src="./app.js"></script>
    <!-- Code injected by live-server -->
    <script type="text/javascript">…</script>
  </body>
</html>
```



```javascript
const heading1 = document.querySelector('.one');
const heading2 = document.querySelector('.two');
const heading3 = document.querySelector('.three')
;

const btn = document.querySelector('.btn');

btn.addEventListener('click', () => {
  addColor(heading1, 1000, 'red')
  .then(() => addColor(heading2, 2000, 'green'))
  .then(() => addColor(heading3, 3000, 'blue'))
  .catch((err)=> console.log(err));
});

const addColor = (element, time, color) => {
  return new Promise((resolve, reject) => {
    if (element) {
      setTimeout(() => {
        element.style.color = color
        resolve();
      }, time);
    } else {
      reject(new Error(`there is no such
      element  ${element}`))
    }
  })
}
```

```
10290 (main)
$ git log --pretty=format:"%h%x09%an%x09%ad%x09%s" --afte
```

Asynchronous Javascript

hello world

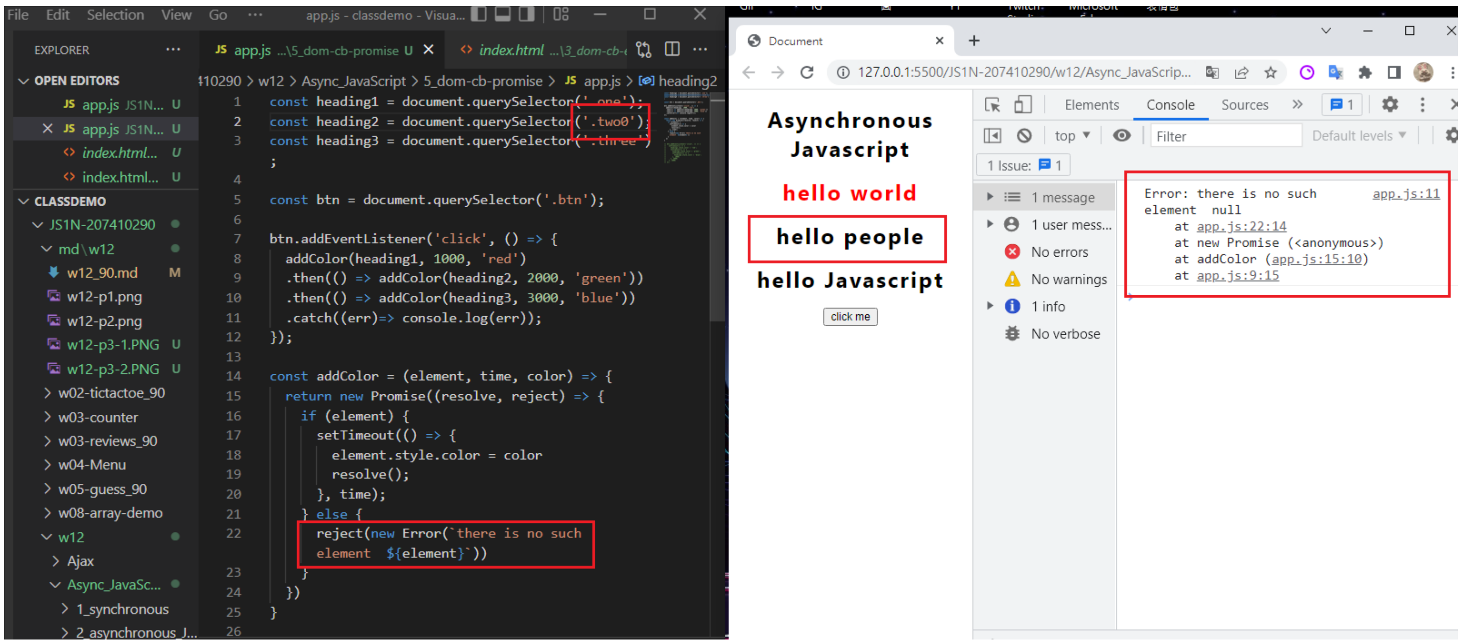
hello people

hello Javascript

click me

```html
<!DOCTYPE html>
<html lang="en"> == $0
  <head>…</head>
  <body>
    <h1>Asynchronous Javascript</h1>
    <h1 class="one" style="color: red;">hello world</h1>
    <h1 class="two" style="color: green;">hello people</h1>
    <h1 class="three" style="color: blue;">hello Javascript</
    <button class="btn">click me</button>
    <script src="./app.js"></script>
    <!-- Code injected by live-server -->
    <script type="text/javascript">…</script>
  </body>
</html>
```

```
element.style {
}
html[Attributes Style] {
```



```javascript
const heading1 = document.querySelector('.one');
const heading2 = document.querySelector('.two0');
const heading3 = document.querySelector('.three')
;

const btn = document.querySelector('.btn');

btn.addEventListener('click', () => {
  addColor(heading1, 1000, 'red')
  .then(() => addColor(heading2, 2000, 'green'))
  .then(() => addColor(heading3, 3000, 'blue'))
  .catch((err)=> console.log(err));
});

const addColor = (element, time, color) => {
  return new Promise((resolve, reject) => {
    if (element) {
      setTimeout(() => {
        element.style.color = color
        resolve();
      }, time);
    } else {
      reject(new Error(`there is no such
      element  ${element}`))
    }
  })
}
```

Asynchronous Javascript

hello world

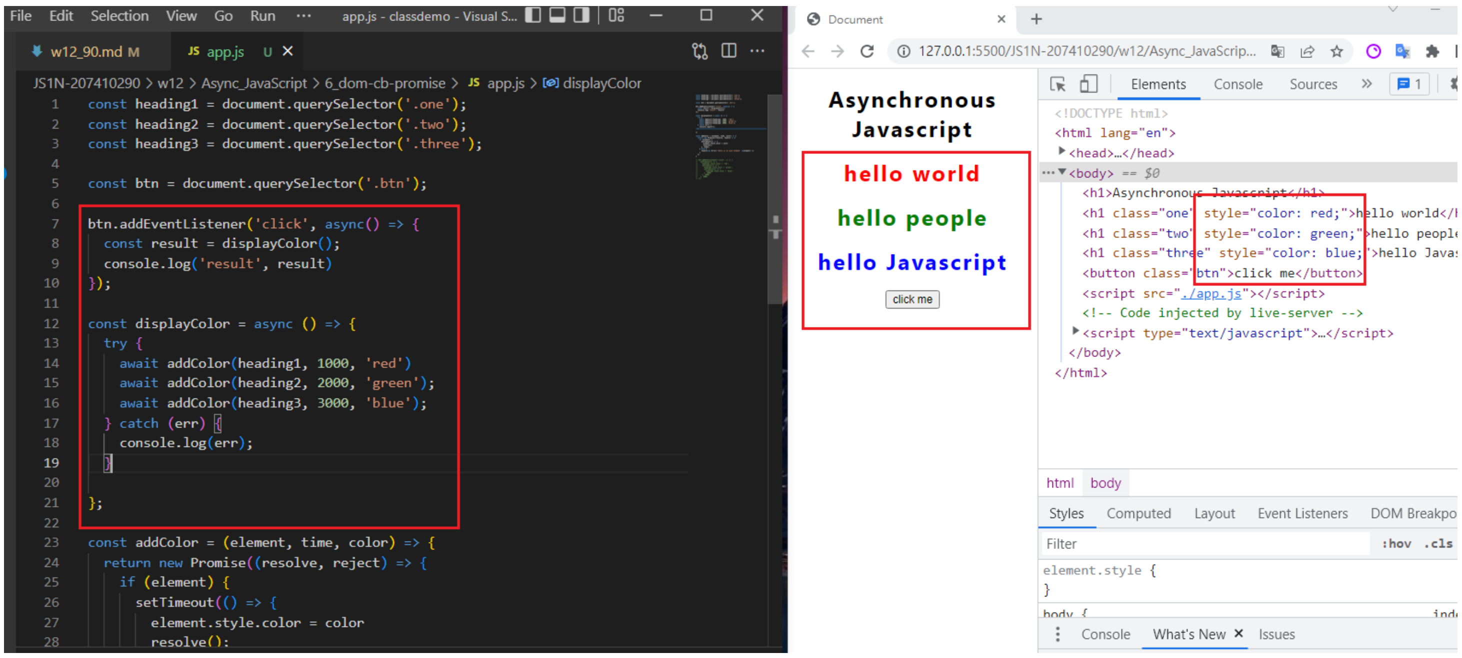
hello people

hello Javascript

click me

```
Error: there is no such                app.js:11
element  null
    at app.js:22:14
    at new Promise (<anonymous>)
    at addColor (app.js:15:10)
    at app.js:9:15
```

w12-P4 use async /await for w12-P3


File Edit Selection View Go Run ··· app.js - classdemo - Visual S...
w12_90.md M
JS app.js U
JS1N-207410290 > w12 > Async_JavaScript > 6_dom-cb-promise > JS app.js > displayColor
Document
127.0.0.1:5500/JS1N-207410290/w12/Async_JavaScrip...
Elements Console Sources
Styles Computed Layout Event Listeners DOM Breakpo
Filter :hov .cls
Console What's New Issues

```javascript
const heading1 = document.querySelector('.one');
const heading2 = document.querySelector('.two');
const heading3 = document.querySelector('.three');

const btn = document.querySelector('.btn');

btn.addEventListener('click', async() => {
  const result = displayColor();
  console.log('result', result)
});

const displayColor = async () => {
  try {
    await addColor(heading1, 1000, 'red')
    await addColor(heading2, 2000, 'green');
    await addColor(heading3, 3000, 'blue');
  } catch (err) {
    console.log(err);
  }

};

const addColor = (element, time, color) => {
  return new Promise((resolve, reject) => {
    if (element) {
      setTimeout(() => {
        element.style.color = color
        resolve();
```

Asynchronous Javascript

hello world

hello people

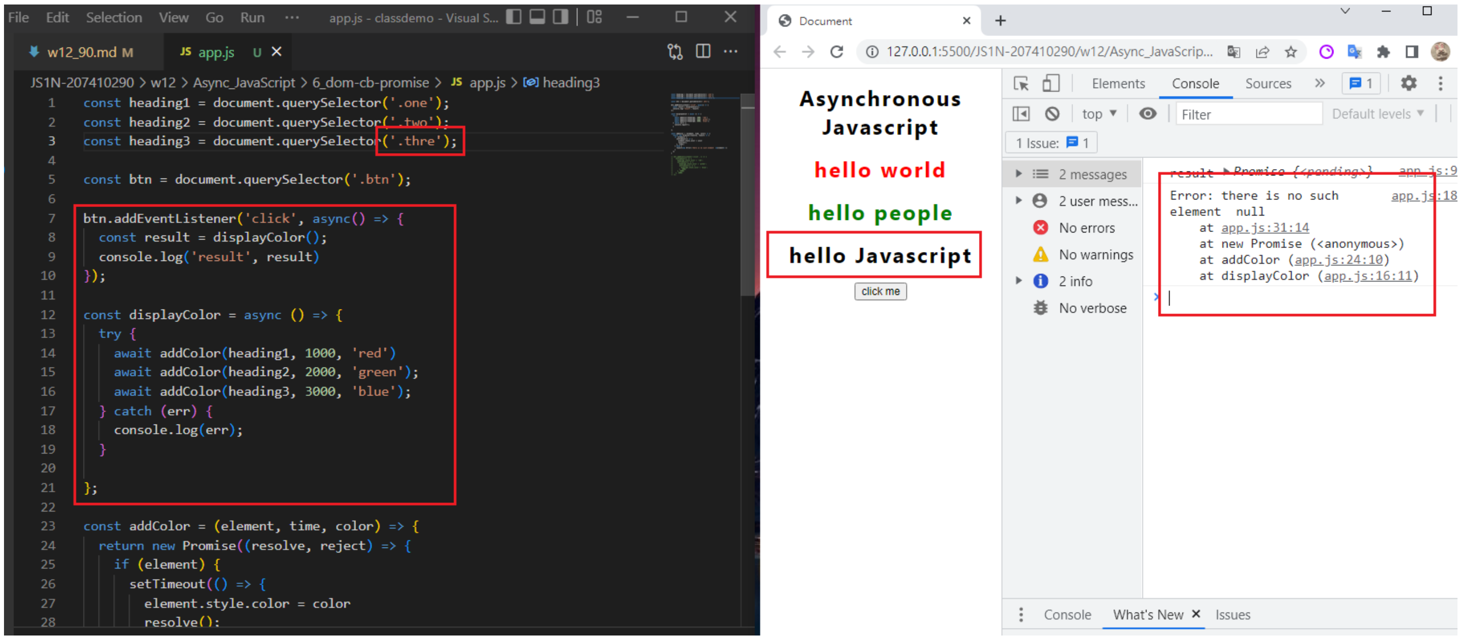
hello Javascript

click me

```html
<!DOCTYPE html>
<html lang="en">
▶<head>…</head>
▼<body> == $0
  <h1>Asynchronous Javascript</h1>
  <h1 class="one" style="color: red;">hello world</h
  <h1 class="two" style="color: green;">hello peopl
  <h1 class="three" style="color: blue;">hello Java
  <button class="btn">click me</button>
  <script src="./app.js"></script>
  <!-- Code injected by live-server -->
  ▶<script type="text/javascript">…</script>
 </body>
</html>
```

html body

```
element.style {
}
body {
```


File Edit Selection View Go Run ··· app.js - classdemo - Visual S...
w12_90.md M
JS app.js U
JS1N-207410290 > w12 > Async_JavaScript > 6_dom-cb-promise > JS app.js > heading3
Document
127.0.0.1:5500/JS1N-207410290/w12/Async_JavaScrip...
Elements Console Sources
top Filter Default levels
1 Issue: 1
2 messages
2 user mess...
No errors
No warnings
2 info
No verbose
Console What's New Issues

```javascript
const heading1 = document.querySelector('.one');
const heading2 = document.querySelector('.two');
const heading3 = document.querySelector('.thre');

const btn = document.querySelector('.btn');

btn.addEventListener('click', async() => {
  const result = displayColor();
  console.log('result', result)
});

const displayColor = async () => {
  try {
    await addColor(heading1, 1000, 'red')
    await addColor(heading2, 2000, 'green');
    await addColor(heading3, 3000, 'blue');
  } catch (err) {
    console.log(err);
  }
};

const addColor = (element, time, color) => {
  return new Promise((resolve, reject) => {
    if (element) {
      setTimeout(() => {
        element.style.color = color
        resolve();
```

Asynchronous Javascript

hello world

hello people

hello Javascript

click me

```
result ▶Promise {<pending>}                 app.js:9
Error: there is no such                    app.js:18
element  null
    at app.js:31:14
    at new Promise (<anonymous>)
    at addColor (app.js:24:10)
    at displayColor (app.js:16:11)
>
```

w12-P5 fetch people.json with 4 data shown on console, change name to people2.json will result in error