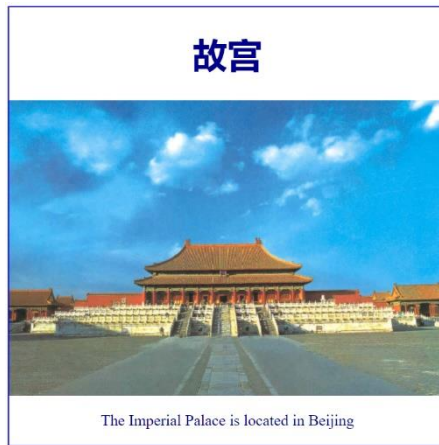


Teddywaddy Code Club

Activity 4b

Visit China



<https://trinket.io/html/ded6cdf90f>

Visit China

This exercise will create a web page and then add Javascript to create a dynamic effect.

Firstly, open the trinket reference shown on the front page (if you haven't already!). This Trinket shows the basic structure of most web pages.

Or if you have VS Code installed, just type in the code.

The image shows a screenshot of the Trinket online code editor. The editor is open to a file named 'index.html'. The code is as follows:

```
1 <html>
2   <head>
3     <title>    </title>
4     <style>
5
6     </style>
7   </head>
8   <body>
9
10
11
12 </body>
13 </html>
14
```

Two callout boxes are present:

- The first callout box points to the opening and closing angle brackets of the `<title>` and `</title>` tags. It contains the text: "The items in <> are known as HTML tags".
- The second callout box points to the opening `<body>` and closing `</body>` tags. It contains the text: "Most tags (but not all) have a start and end, like the <body> and </body> pair."

Let's start completing the HTML component of this web page.

The image shows a screenshot of the Trinket IDE interface. At the top, there is a navigation bar with the Trinket logo, an 'Autorun' button, and a dropdown menu. Below this, a file explorer shows 'index.html' selected. The main area displays the following HTML code:

```
1 <html>
2   <head>
3     <title> </title>
4     <style>
5
6   </style>
7 </head>
8 <body>
9
10
11
12 </body>
13 </html>
14
```

Three callout boxes provide instructions for completing the code:

- Add a title**
`<title>Visit China</title>`
- Add a Heading**
`<h1>Imperial Palace</h1>`
- Add an image**
``
Notice that there is no end `` tag.

Don't worry that the image is too large at this stage.

Mostly, the content of a web page, so writing, images and so on goes within the <body> part of the html file.

How that content is formatted, or styled, is described in the <style> section, which is part of the <head> section in a html file.

Add some style.



```
1 <html>
2 <head>
3 <title>Visit China</title>
4 <style>
5
6 </style>
7 </head>
8 <body>
9
10 <h1>Imperial Palace</h1>
11 
12
13 </body>
14 </html>
15
```

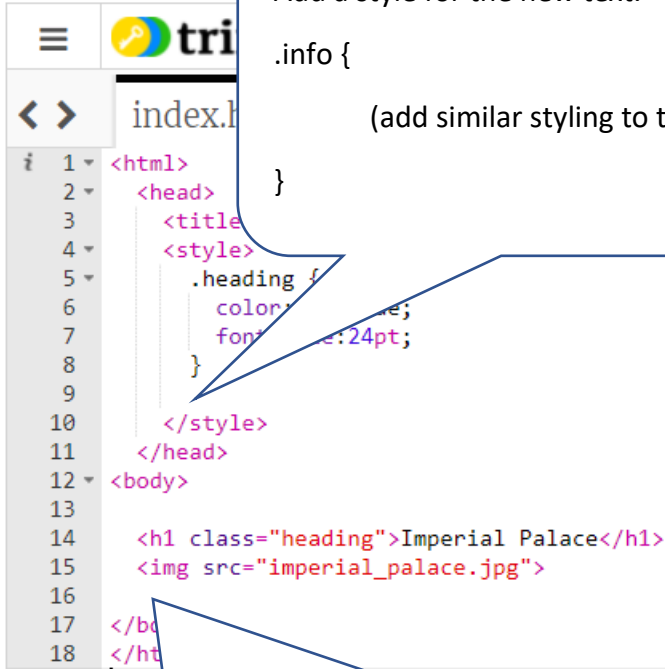
Add a style for the heading

```
. heading {
  color: darkblue;
  font-size: 24pt;
}
```

In order to connect the html element to the style the name of the style needs to be listed in the tag.

```
<h1 class="heading">
```

Add some more text to the web page.



```
1 <html>
2 <head>
3 <title>
4 <style>
5 .heading {
6 color: #e91e63;
7 font-size: 24pt;
8 }
9
10 </style>
11 </head>
12 <body>
13
14 <h1 class="heading">Imperial Palace</h1>
15 
16
17 </body>
18 </html>
```

Add a style for the new text.

```
.info {
    (add similar styling to the heading)
}
```

Add a Paragraph to the bottom.

```
<p class="info">The Imperial Palace is located in Beijing</p>
```

The image is too large and the writing isn't centered. To fix these issues a new tag is needed around all the content so far - the `<div>` and `</div>` pair.


```
1 <html>
2 <head>
3 <title>V
4 <style>
5 .heading {
6 color:darkred;
7 font-size:3
8 }
9 .info {
10 color:blue;
11 font-size:24pt;
12 }
13 </style>
14 </head>
15 <body>
16
17 <h1 class="heading">Imperial Palace</h1>
18 
19 <p class="info">The Imperial Palace is located in Beijing</p>
20
21 </body>
22 </html>
```

`<div class="container">`

Remember – “container” is the style name

`</div>`

Next add the new `.container` style for this `<div>`. See next page.



```
1 <html>
2 <head>
3   <title>Visit China</title>
4   <style>
5
6     .heading {
7       color:darkblue;
8       font-size:36pt;
9     }
10    .info {
11      color: darkblue;
12      font-size:24pt;
13    }
14  </style>
15 </head>
16 <body>
17   <div class="container">
18     <h1 class="heading">Imperial Palace</h1>
19     
20     <p class="info">The Imperial Palace is located in Beijing</p>
21   </div>
22 </body>
23 </html>
```

.container {
width: 600px;
margin: auto;
border: blue solid 2px;
text-align: center;
}

Finally there needs to be a style applied to the image.

The screenshot shows a code editor interface with the following code:

```
1 <html>
2 <head>
3 <title>Visit Chin
4 <style>
5   .container {
6     width: 600px;
7     margin: auto;
8     border: blue
9     text-align: c
10  }
11  .heading {
12    color:darkb
13    font-size
14  }
15  .info {
16    color:darkblu
17    font-size:24pt;
18  }
19
20 </style>
21 </head>
22 <body>
23 <div class="container">
24 <h1 class="heading">Imperial Palace</h1>
25 
26 <p class="info">The Imperial Palace is located in Beijing</p>
27 </div>
28 </body>
29 </html>
```

A callout box points to the CSS rule for the `.image` class:

```
.image {
  width: 100%;
}
```

Note that this means 100% of the container, which is 600px, so the image should perfectly fit the container.

Another callout box points to the `` tag in the HTML code:

```

```

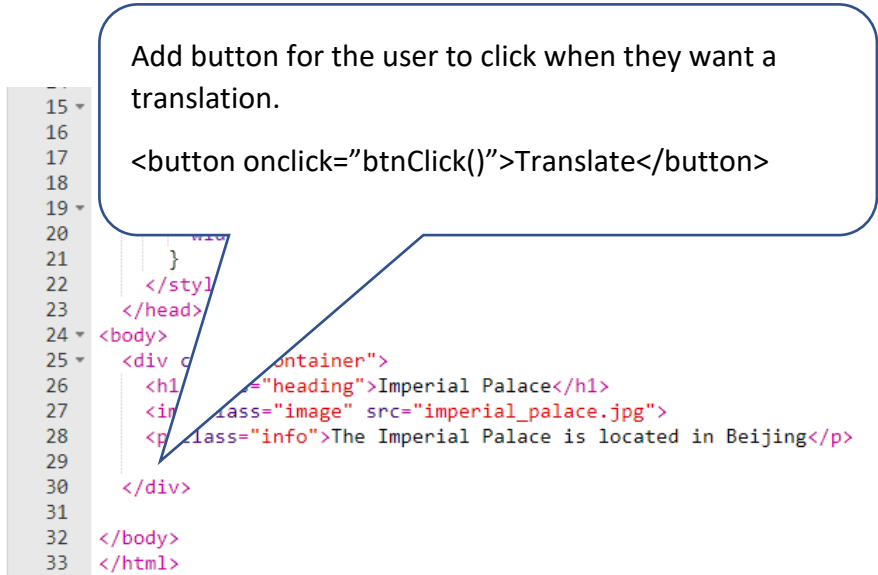
This should now have the web page looking much better.

The next requirement is for the user to be able to change the heading from English to Chinese. This will require some JavaScript coding.

Add button for the user to click when they want a translation.

```
<button onclick="btnClick()">Translate</button>
```

```
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
```



The `onclick="btnClick()"` joins together an event called `onclick` and a JavaScript function called `btnClick()`.

`btnClick()` is the name for some JavaScript code that must be written – it will set out what to do when the user clicks the button.

Just like the class attribute connects HTML to styles, The id attribute connects HTML to JavaScript

class is for applying styles

id is for identification in JavaScript coding

```
15  .info {
16      color: darkblue;
17      font-size:24pt;
18  }
19
20  <h1 id="txt" class="heading">
21
22  </style>
23  </head>
24  <body>
25  <div class="container">
26      <h1 class="heading">Imperial Palace</h1>
27      
28      <p class="info">The Imperial Palace is located in Beijing</p>
29      <button onclick="btnClick()">Translate</button>
30  </div>
31
32  </b>
33  <
```

`<h1 id="txt" class="heading">`

Add the JavaScript just before the `</body>`

```
<script>
    function btnClick() {
        document.getElementById("txt").innerHTML = " 故宫";
    }
</script>
```

To get the Chinese characters go to

<http://www.teddywaddy.com.au/chinese.html>

If you can't get the Chinese characters in electronic format, you can use the following code.

```
function btnClick() {  
    var chinese = String.fromCharCode(25925) + String.fromCharCode(23467);  
    document.getElementById("txt").innerHTML = chinese;  
}
```

Test out the button. Note that it doesn't go back to English. Refresh the browser to get the display back to the start.

Extension

Add another button that displays the heading in English.

You will need to do the following.

1. Add another button (with different text and onclick function). Like this.

```
<button onclick="chinese()">Chinese</button>
```

```
<button onclick="english()">English</button>
```

2. Add another function like this:

```
function english() {  
    document.getElementById("txt").innerHTML = "Imperial Palace";  
}
```

3. Create and apply a style to make the buttons look better.