

Jobsheet-6: JQuery and Bootstrap


Muhammad Tegar Hibatulloh

2341720221 / 22 / 2I

Link Github:

<https://github.com/Garrss/Desain-dan-Pemograman-Web/tree/main/week6>

Practical Section 1: Preparing to Use JQuery

 jquery-3.7.1.min.js

03/10/2024 8:16

JSFile

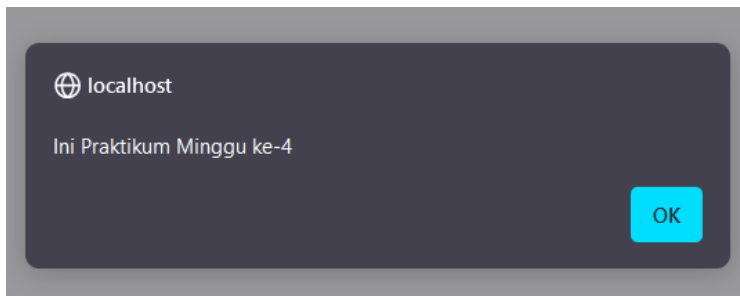
86 KB

```
week6 > JQquery.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Praktikum JQuery</title>
7      <script src="jquery-3.7.1.min.js"></script>
8  </head>
9  <body>
10
11 </body>
12 </html>
```

Practical Section 2.Document Ready Function

```
week6 > ready.html > html > body > input#button1
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <script type="text/javascript" src="jquery-3.7.1.min.js"></script>
5      <script>
6          $(document).ready(function () {
7              $("#button").click(function () {
8                  alert("Ini Praktikum Minggu ke-4");
9              });
10         });
11     </script>
12 </head>
13 <body>
14     <input type="button" id="button1" value="Klik Saya!" />
15 </body>
16 </html>
17
```

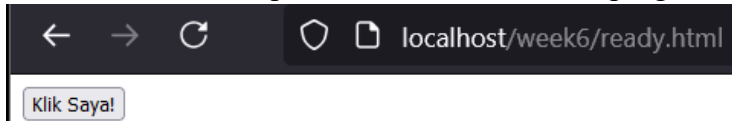
After the program is run, click the "Click Me!" button, then observe the display on the browser.



Next, modify the file by removing the program code on lines 6 and 10, so that the program code looks like this:

```
week6 > ready.html > html > head > script
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <script type="text/javascript" src="jquery-3.7.1.min.js"></script>
5      <script>
6        $("#button1").click(function () {
7          alert("Ini Praktikum Minggu ke-4");
8        });
9      </script>
10   </head>
11   <body>
12     <input type="button" id="button1" value="Klik Saya!" />
13   </body>
14 </html>
15
```

Save the file. Then open a browser and run the program code by typing



1. From your observations, what do you understand from the document ready() function? Record the results of your observations and explanations of your understanding below (Question No. 1)
 - First Code (with \$(document).ready(function() {...}));:
 - The jQuery \$(document).ready() function is used in this code to ensure that the entire HTML document is fully loaded and parsed by the browser before any jQuery code (such as event binding) is executed.
 - The click() function that triggers an alert is enclosed within the ready() function, meaning it will only run after the document is ready.
 - Second Code (without \$(document).ready()):
 - In this code, there is no \$(document).ready() wrapper, so the click() function is directly placed in the <head> section.
 - If this script runs before the HTML is fully loaded (i.e., before the button is created in the DOM), the event handler may not be attached properly. This could lead to the button not responding to clicks as expected.

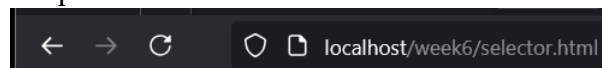
Practical Section 3. Selector

2. Record and explain your observations after running the program code above. (Question No. 2)

Code:

```
week6 > <> selector.html > html > head > body
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <head>
5          <script src="jquery-3.7.1.min.js"></script>
6          <script>
7              $(document).ready (function(){
8                  $("button").click(function(){
9                      $("h2").hide();
10                     $(".subjudul").hide();
11                     $("#paragraf").hide();
12                     $("#paragraf.dua").hide();
13                     $("div p.paragraf").hide();
14                 });
15             });
16         </script>
17     </head>
18     <body>
19         <h2> Ini adalah Judul Bab </h2>
20         <h3 class="subjudul"> Ini adalah Sub judul Bab </h3>
21         <p id="paragraf"> Ini adalah paragraf pertama </p>
22         <p id="paragraf" class="dua"> Ini adalah paragraf kedua </p>
23         <div>
24             <p class="paragraf">
25                 Ini adalah paragraf ketiga
26             </p>
27         </div>
28         <p> Ini paragraf yang akan tetap ditampilkan </P>
29         <button> Klik Saya! </button>
30     </body>
31 </body>
32 </html>
```

Output:



Ini adalah Judul Bab

Ini adalah Sub judul Bab

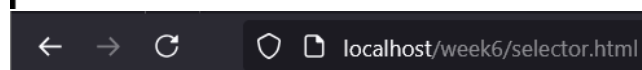
Ini adalah paragraf pertama

Ini adalah paragraf kedua

Ini adalah paragraf ketiga

Ini paragraf yang akan tetap ditampilkan

Klik Saya!



Ini paragraf yang akan tetap ditampilkan

Klik Saya!

The code displays a webpage with headings, paragraphs, and a button. When the button is clicked, several elements are hidden: the `<h2>` heading, the `<h3>` subheading, two paragraphs (the first with `id="paragraf"` and a paragraph inside a `<div>`), but one paragraph without specific markup remains visible.

3. In the program code in step number 1, there are some jQuery selector writings. Write down and explain what jQuery Selector is in your code. (Question No. 3)
- Tag Selector (`$("#h2")`): Selects all `<h2>` elements.
 - Class Selector (`$(".subjudul")`): Selects elements with the class `subjudul`.
 - ID Selector (`$("#paragraf")`): Selects elements with the `id="paragraf"`.
 - ID-Class Combination Selector (`$("#paragraf.dua")`): Incorrectly combines an ID and class; it should be fixed.
 - Descendant Selector (`$("#div p.paragraf")`): Selects paragraphs with the class `paragraf` that are descendants of a `<div>`.

Each selector is used to target specific elements, which are then hidden using the `.hide()` function.

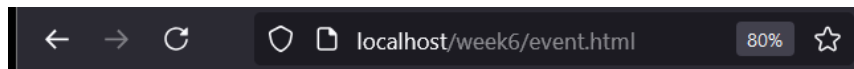
Practical Section 4. Events

4. Record and describe the changes that occurred based on your observations (Question No.4)

Code:

```
week6 > < event.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <style>
5        #paragraf {
6          width: 600px;
7          height: 80px;
8          background-color: pink;
9          line-height: 80px;
10         text-align: center;
11         font-size: 30px;
12       }
13     </style>
14     <script src="jquery-3.7.1.min.js"></script>
15     <script>
16       $(document).ready(function () {
17         $("#paragraf").click(function () {
18           $("#paragraf").css("color", "white");
19         });
20         $("#paragraf").mouseover(function () {
21           $("#paragraf").css("background-color", "silver");
22         });
23         $("#paragraf").mouseout(function () {
24           $("#paragraf").css("background-color", "blue");
25         });
26         $("#paragraf").dblclick(function () {
27           $("#paragraf").css("border", "solid 3px black");
28         });
29       });
30     </script>
31   </head>
32   <body>
33     <p id="paragraf">Praktikum Minggu ke-4</p>
34   </body>
35 </html>
36
```

Output:



Praktikum Minggu ke-4

- Click: The text color changes to white.
- Mouseover (Hover): The background color changes to silver.
- Mouseout (Mouse leaves): The background color changes to blue.
- Double-click: A solid 3px black border appears around the paragraph.

Practical Section 5. Hide and Show Effect

5. Record and describe the changes that occurred based on your observations (Question No.5)

Code:

```
week6 > < hideshow.html > html > head > script > ready() callback > click() callback
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <link rel="stylesheet" type="text/css" href="style.css" />
5 <script src="jquery-3.7.1.min.js"></script>
6 <script>
7     $(document).ready(function () {
8         $(".tombol1").click(function () {
9             $(".div").hide();
10        });
11        $(".tombol2").click(function () {
12            $(".div").show();
13        });
14        $(".tombol3").click(function () {
15            $("#div1").fadeOut();
16            $("#div2").fadeOut("slow");
17            $("#div3").fadeOut(3000);
18        });
19        $(".tombol4").click(function () {
20            $(".div1").fadeIn();
21            $(".div2").fadeIn("slow");
22            $(".div3").fadeIn(3000);
23        });
24        $(".tombol5").click(function () {
25            $(".div1").fadeTo("slow", 0.15);
26            $(".div2").fadeTo("slow", 0.4);
27            $(".div3").fadeTo("slow", 0.7);
28        });
29    });
30 </script>
31 </head>
32 <body>
33 <p>jQuery Effect</p>
34 <div id="div1" class="kotak"></div>
35 <br /><br />
36 <div id="div2" class="kotak"></div>
37 <br /><br />
38 <div id="div3" class="kotak"></div>
39 <br /><br />
40 <button class="tombol1">Sembunyikan</button>
41 <button class="tombol2">Tampilkan</button>
42 <button class="tombol3">Fade Out</button>
43 <button class="tombol4">Fade In</button>
44 <button class="tombol5">Fade To</button>
45 </body>
46 </html>
47
```

```

week6 > # style.css > #div2
1  .kotak {
2      width: 80px;
3      height: 80px;
4  }
5
6  #div1 {
7      background-color: red;
8  }
9
10 #div2 {
11     background-color: green;
12 }
13
14 #div3 {
15     background-color: blue;
16 }
17

```

- Sembunyikan: All boxes are hidden.
- Tampilkan: All boxes are shown again.
- Fade Out: The boxes fade out at different speeds (instant, slow, 3 seconds).
- Fade In: The boxes fade back in at different speeds, but Box 1 does not fade in due to an incorrect selector.
- Fade To: The boxes fade to specific opacity levels (15%, 40%, 70%).
- The selector \$("div1") in the Fade In function should be corrected to \$("#div1") for Box 1 to fade in properly. After fixing this, the box will fade in as expected.

Practical Section 6. Slide Effect

- Record and explain the changes that occurred based on your observations (Question No. 6)

Code:

```

week6 > slide1.html > html > body > div#kotak2
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <link rel="stylesheet" type="text/css" href="styleSlide.css" />
5      <script src="jquery-3.7.1.min.js"></script>
6      <script>
7          $(document).ready(function () {
8              $("#flip").click(function () {
9                  $("#kotak2").slideUp("slow");
10             });
11         });
12     </script>
13 </head>
14 <body>
15     <div id="flip">Klik untuk Efek Slide</div>
16     <div id="kotak2">Selamat Pagi</div>
17 </body>
18 </html>
19

```

```

week6 > # styleSlide.css > ...
1  #kotak2,
2  #flip {
3      padding: 5px;
4      text-align: center;
5      background-color: #a7dbd8;
6      border: solid 2px #e0e4cc;
7      border-radius: 5px;
8  }
9
10 #kotak2 {
11     padding: 50px;
12 }
13

```

Output:



- Before Click: Both #flip and #kotak2 are visible with their respective text and styles.
- After Click: #kotak2 slides up and hides with a slow animation, while #flip remains visible.

7. Record and describe the changes that occur based on your observations (Question No. 7)

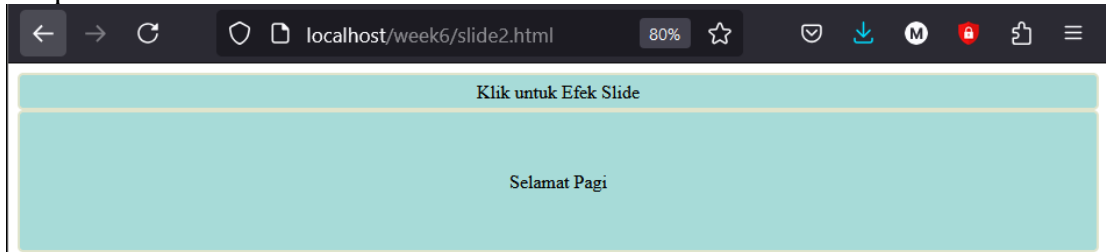
Code:

```

week6 > <> slide2.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3      <head>
4          <link rel="stylesheet" type="text/css" href="styleSlide.css" />
5          <script src="jquery-3.7.1.min.js"></script>
6          <script>
7              $(document).ready(function () {
8                  $("#flip").click(function () {
9                      $("#kotak2").slideDown("slow");
10                 });
11             });
12         </script>
13     </head>
14     <body>
15         <div id="flip">Klik untuk Efek Slide</div>
16         <div id="kotak2" style="display: none">Selamat Pagi</div>
17     </body>
18 </html>
19

```

Output:



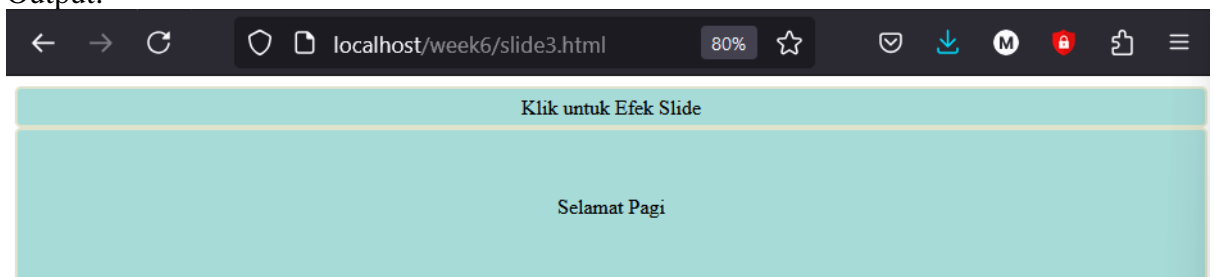
- Before Click: #flip is visible with the prompt, and #kotak2 is hidden.
- After Click: #kotak2 slides down and becomes visible with the text "Selamat Pagi". The slide-down animation occurs slowly.

8. Record and describe the changes that occurred based on your observations (Question No. 8)

Code:

```
week6 > slide3.html > html > body > div#kotak2
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <link rel="stylesheet" type="text/css" href="styleSlide.css" />
5     <script src="jquery-3.7.1.min.js"></script>
6     <script>
7       $(document).ready(function () {
8         $("#flip").click(function () {
9           $("#kotak2").slideToggle("slow");
10        });
11      });
12    </script>
13  </head>
14  <body>
15    <div id="flip">Klik untuk Efek Slide</div>
16    <div id="kotak2">Selamat Pagi</div>
17  </body>
18 </html>
19
```

Output:



- Before Click: Both #flip and #kotak2 are visible.
- After Click:
 - If #kotak2 was visible, it slides up and becomes hidden.
 - If it were hidden initially (if modified), it would slide down and become visible.
- The toggle action creates a smooth animation effect, enhancing user interaction.

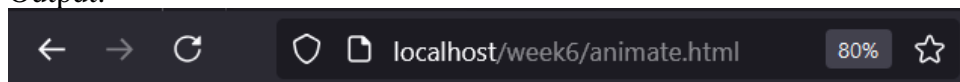
Practical Section 7. Animation

9. Record and describe the changes that occurred based on your observations (Question No. 9)

Code:

```
week6 > <> animate.html > html > body > div.box
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <link rel="stylesheet" type="text/css" href="styleAnimate.css" />
5      <script src="jquery-3.7.1.min.js"></script>
6      <script>
7        $(document).ready(function () {
8          $("button").click(function () {
9            $("div").animate({ left: 300 });
10          });
11        });
12      </script>
13    </head>
14    <body>
15      <button type="button">Start Animation</button>
16      <div class="box"></div>
17    </body>
18  </html>
19
```

Output:



- Before Click: The div element (box) is in its original position.
- After Click: The div element slides 300 pixels to the right due to the animation of the left property.
- The animation is triggered by clicking the button, creating a visual movement of the div on the page.

10. Record and explain the changes that occur based on your observations (Question No. 10)

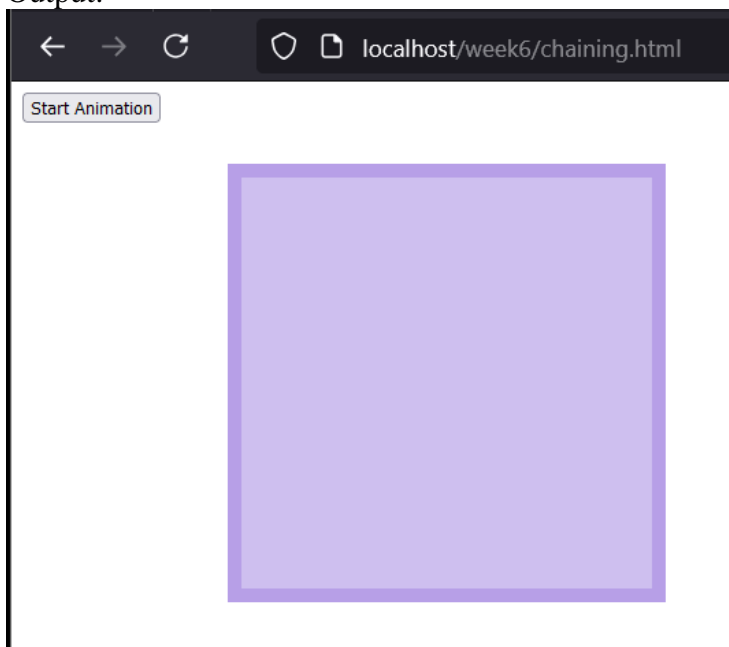
Code:

```

week6 > < chaining.html > html > body > div.box
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  <link rel="stylesheet" type="text/css" href="styleAnimate.css" />
5  <script src="jquery-3.7.1.min.js"></script>
6  <script>
7  $(document).ready(function () {
8      $("button").click(function () {
9          $(".box")
10             .animate({ width: "300px" })
11             .animate({ height: "300px" })
12             .animate({ marginLeft: "150px" })
13             .animate({ borderWidth: "10px" })
14             .animate({ opacity: 0.5 });
15      });
16  });
17 </script>
18 </head>
19 <body>
20 <button type="button">Start Animation</button>
21 <div class="box"></div>
22 </body>
23 </html>
24

```

Output:



- Before Click: The box has its original width, height, position, border, and full opacity.
- After Click:
 - The box becomes a 300px by 300px square.
 - It moves 150px to the right.
 - The border width increases to 10px.
 - The opacity decreases to 0.5, making the box semi-transparent.
- The animations occur one after the other in sequence, creating a smooth transition between the changes.

Practical Section 8. JQuery DOM Manipulation

11. Record your observations and explain the differences between the `get text()`, `html()`, and `val()` methods (Question No. 11)

Code:

```

week6 > <> get.html > html > body > p
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <script src="jquery-3.7.1.min.js"></script>
5      <script>
6        $(document).ready(function () {
7          $("#btn1").click(function () {
8            alert($("#test1").text());
9          });
10         $("#btn2").click(function () {
11           alert($("#test2").html());
12         });
13         $("#btn3").click(function () {
14           alert($("#test3").val());
15         });
16       });
17     </script>
18   </head>
19   <body>
20     <p id="test1">This is the first paragraph.</p>
21     <p id="test2">This is <b> another </b> paragraph.</p>
22
23     <p>
24       Your Name: <input type="text" id="test3" /></p>
25
26     <button id="btn1">Get Text</button>
27     <button id="btn2">Get HTML</button>
28     <button id="btn3">Get Value</button>
29   </p>
30 </body>
31 </html>
32

```

Output;

← → ↻ 🛡 📄 localhost/week6/get.html

This is the first paragraph.

This is **another** paragraph.

Your Name: /p>

- text(): Retrieves only the plain text of an element, ignoring any HTML tags.
- html(): Retrieves the HTML content, including tags, within an element.
- val(): Retrieves the value of form elements such as inputs, typically user-entered data.

12. Record your observations and explain the differences between the set text(), html(), and val() methods (Question No. 12)

Code:

```

week6 > <> get.html > html > body
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <script src="jquery-3.7.1.min.js"></script>
5      <script>
6        $(document).ready(function () {
7          $("#btn1").click(function () {
8            $("#test1").text("<b>Hello World</b>");
9          });
10         $("#btn2").click(function () {
11           $("#test2").html("<b>Hello World!</b>");
12         });
13         $("#btn3").click(function () {
14           $("#test3").val("Polinema");
15         });
16       });
17     </script>
18   </head>
19   <body>
20     <p id="test1">This is the first paragraph.</p>
21     <p id="test2">This is <b> another </b> paragraph.</p>
22
23     <p>
24       Your Name: <input type="text" id="test3" /></p>
25
26     <button id="btn1">Set Text</button>
27     <button id="btn2">Set HTML</button>
28     <button id="btn3">Set Value</button>
29   </p>
30 </body>
31 </html>
32

```

Output:

Hello World

Hello World!

Your Name:

- .text() works with plain text only and ignores any HTML tags.
- .html() allows you to manipulate HTML content, interpreting tags correctly.
- .val() is specifically for setting or retrieving values from form elements.

13. Record and explain your observations (Question No. 13)

Code:

```

week6 > > addremove.html > html > head > script > ready() callback > click() callback
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <link rel="stylesheet" type="text/css" href="styleTable.css" />
5      <script src="jquery-3.7.1.min.js"></script>
6      <script>
7        $(document).ready(function () {
8          //menambah baris
9          $(".add-row").click(function () {
10             var name = $("#name").val();
11             var email = $("#email").val();
12             var markup =
13               "<tr><td><input type='checkbox' name='record'></td><td>" +
14               name +
15               "</td><td>" +
16               email +
17               "</td></tr>";
18             $("table tbody").append(markup);
19           });
20           //menghapus baris yang di centang
21           $(".delete-row").click(function () {
22             $("table tbody")
23               .find('input[name="record"]')
24               .each(function () {
25                 if ($(this).is(":checked")) {
26                   $(this).parents("tr").remove();
27                 }
28               });
29           });
30         });
31       </script>
32     </head>
33     <body>
34       <form>
35         <input type="text" id="name" placeholder="Name" />
36         <input type="text" id="email" placeholder="Email Address" />
37         <input type="button" class="add-row" value="Add Row" />
38       </form>
39       <table>
40         <thead>
41           <tr>
42             <th>Select</th>
43             <th>Name</th>
44             <th>Email</th>
45           </tr>
46         </thead>
47         <tbody>
48           <tr>
49             <td><input type="checkbox" name="record" /></td>
50             <td>Tony Stark</td>
51             <td>tonystark@gmail.com</td>
52           </tr>
53         </tbody>
54       </table>
55       <button type="button" class="delete-row">Delete Row</button>
56     </body>
57   </html>
58

```

Output:

tegar 07mtegar@gmail.com Add Row

Select	Name	Email
<input checked="" type="checkbox"/>	Tony Stark	tonystark@gmail.com
<input type="checkbox"/>	tegar	07mtegar@gmail.com

Delete Row

Observations:

- Adding Rows: When the "Add Row" button is clicked, the entered "Name" and "Email" values are used to create a new table row with a checkbox. This row is appended to the table.

- Deleting Rows: When the "Delete Row" button is clicked, all rows with checked checkboxes are removed from the table.
- Functionality: The code utilizes jQuery to add rows using the .append() method and delete rows by identifying checked checkboxes and removing their parent <tr> elements.

Behavior:

- Rows can be added and removed dynamically based on user interaction.

Practical Section 9. JQuery CSS Manipulation

14. Record and explain your observations (Question No. 14)

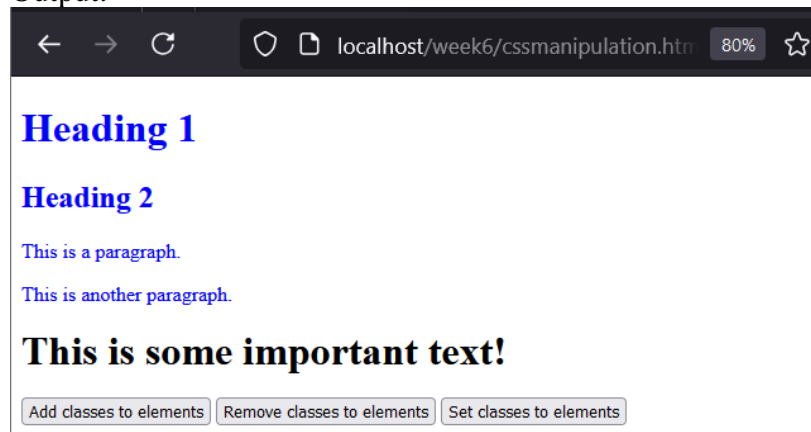
Code:

```

week6 > <cssmanipulation.html> <html> <head> <script> <ready() callback> <click() callback>
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <link rel="stylesheet" type="text/css" href="styleCSS.css" />
5 <script src="jquery-3.7.1.min.js"></script>
6 <script>
7 $(document).ready(function () {
8     $(".btn1").click(function () {
9         $("h1, h2, p").addClass("blue");
10        $("div").addClass("important");
11    });
12    $(".btn2").click(function () {
13        $("h1, h2, p").removeClass("blue");
14    });
15    $(".btn3").click(function () {
16        $("h1, h2, p, div").css({
17            "background-color": "yellow",
18            "font-size": "100%",
19        });
20    });
21 });
22 </script>
23 </head>
24 <body>
25 <h1>Heading 1</h1>
26 <h2>Heading 2</h2>
27 <p>This is a paragraph.</p>
28 <p>This is another paragraph.</p>
29 <div>This is some important text!</div>
30 <br />
31
32 <button class="btn1">Add classes to elements</button>
33 <button class="btn2">Remove classes to elements</button>
34 <button class="btn3">Set classes to elements</button>
35 </body>
36 </html>
37

```

Output:



- .addClass(): Adds the specified class (.blue or .important) to the selected elements, applying the associated CSS rules.
- .removeClass(): Removes the specified class (.blue) from the selected elements, removing the associated CSS rules.
- .css(): Directly applies inline CSS styles (such as background-color and font-size) to the selected elements, overriding any existing styles.

Practical Section 10. Create a Slide Show

15. Record and explain your observations (Question No. 15)

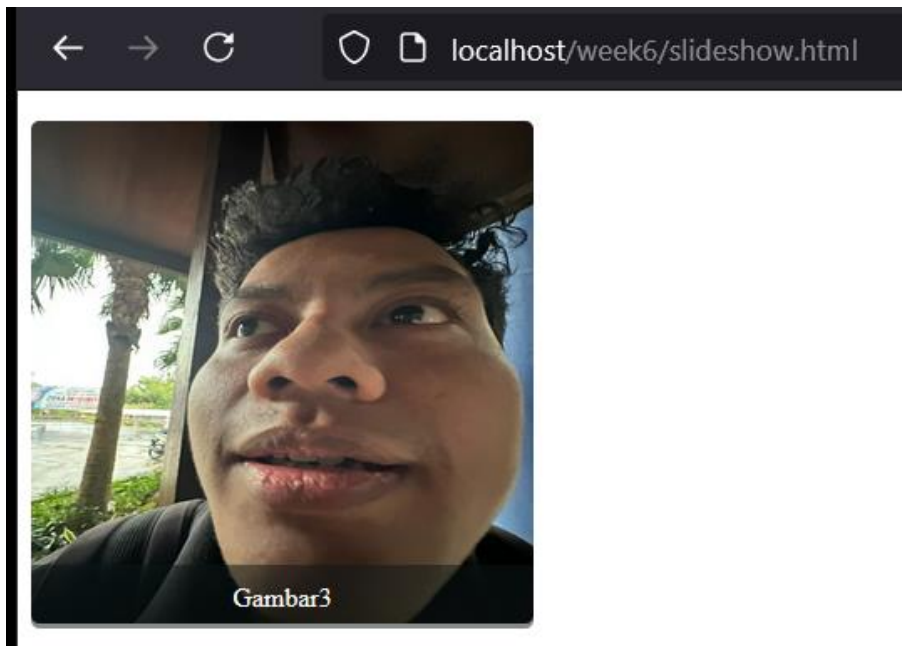
Code:

```

week6 > > slideshow.html > html > body > div#slider > img#sliderImage2
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  <link rel="stylesheet" type="text/css" href="styleSlideShow.css" />
5  <script src="jquery-3.7.1.min.js"></script>
6  <script>
7      var i = 0;
8      $(document).ready(function () {
9          $(".slidertitle, #slider img").hide();
10         showNextImage();
11         setInterval("showNextImage()", 3000);
12     });
13
14     function showNextImage() {
15         i++;
16         $("#sliderImage" + i)
17             .appendTo("#slider")
18             .fadeIn(1100)
19             .delay(1100)
20             .fadeOut(1100);
21         $("#title" + i)
22             .appendTo("#slider")
23             .fadeIn(1100)
24             .delay(1100)
25             .fadeOut(1100);
26         if (i == 3) {
27             i = 0;
28         }
29     }
30 </script>
31 </head>
32 <body>
33     <div id="slider">
34         
35         <div class="slidertitle" id="tittle1">Gambar1</div>
36
37         
38         <div class="slidertitle" id="title2">Gambar2</div>
39
40         
41         <div class="slidertitle" id="title3">Gambar3</div>
42     </div>
43 </body>
44 </html>
45

```

Output:



- The slideshow works as intended, displaying one image and its title at a time, smoothly transitioning between them every 3 seconds.
- The fadeIn, fadeOut, and delay effects control the timing of the transitions, providing a smooth and visually appealing effect.
- The images and titles are displayed in sequence, and the slideshow loops indefinitely.

Practical Section 11. JQuery UI Plugin

16. Record and explain your observations (Question No. 15)

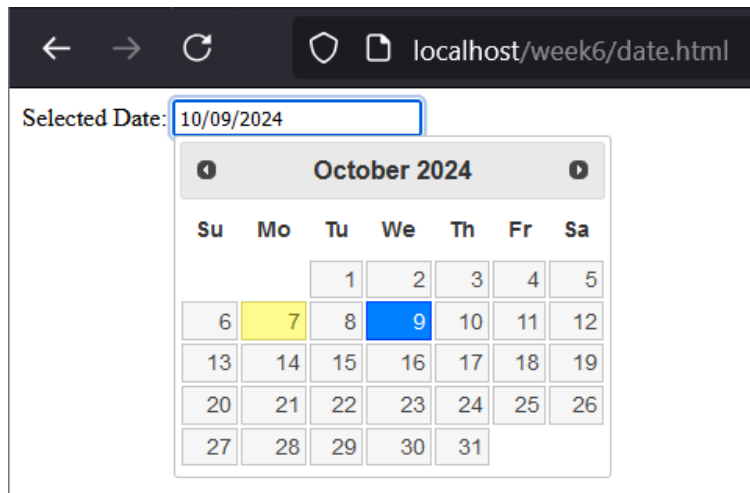
Code:

```

week6 > > date.html > html > head > script
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <link
5        rel="stylesheet"
6        href="jquery-ui-1.14.0/jquery-ui-1.14.0/jquery-ui.js"
7      />
8      <script src="jquery-3.7.1.min.js"></script>
9      <script src="jquery-ui-1.14.0/jquery-ui-1.14.0/jquery-ui.js"></script>
10     <script>
11       $(function () {
12         $("#date_ex").datepicker();
13       });
14     </script>
15   </head>
16   <body>
17     <div>Selected Date: <input type="text" id="date_ex" /></div>
18   </body>
19 </html>
20

```

Output:



- The code works as expected, initializing a jQuery UI date picker on the input field.
- However, the missing or incorrect link to the jQuery UI CSS file could affect the visual appearance of the date picker.

17. Record and explain your observations (Question No. 16)

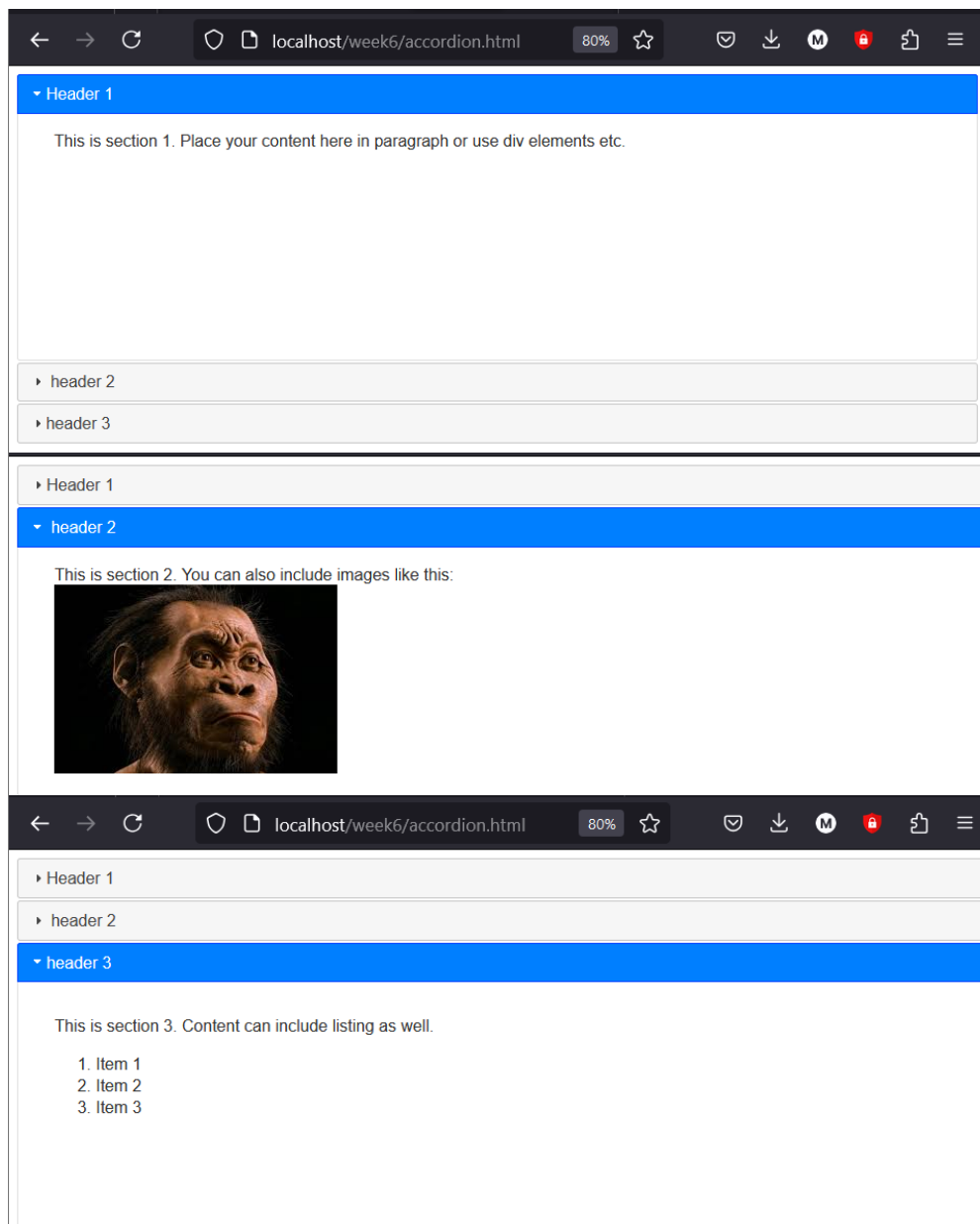
Code:

```

week6 > accordion.html > html > body > div#jQuery_accordion
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <link rel="stylesheet" href="jquery-ui-1.14.0/jquery-ui-1.14.0/jquery-ui.css">
5   <script src="jquery-3.7.1.min.js"></script>
6   <script src="jquery-ui-1.14.0/jquery-ui-1.14.0/jquery-ui.js"></script>
7   <script>
8     $(function() {
9       $("#jQuery_accordion").accordion();
10    });
11  </script>
12 </head>
13 <body>
14   <div id="jQuery_accordion">
15     <h3>Header 1</h3>
16     <p> This is section 1.
17       Place your content here in paragraph or use div elements etc. </p>
18     <h3> header 2</h3>
19     <p> This is section 2. You can also include images like this:
20       <br> </p>
21     <h3>header 3</h3>
22     <div>
23       <p>This is section 3. Content can include listing as well.
24         <ol>
25           <li>Item 1</li>
26           <li>Item 2</li>
27           <li>Item 3</li>
28         </ol>
29       </p>
30     </div>
31   </div>
32 </body>
33 </html>

```

Output:



- The jQuery UI accordion transforms the headers and associated content into a collapsible/expandable structure. The sections toggle based on user interaction with smooth animations.
- The code is structured properly, with each section containing relevant content (text, image, and lists), offering a practical user interface for displaying large amounts of content in a compact, organized manner.

18. Record and explain your observations (Question No. 17)

Code:

```

week6 > <> test-content.html > html > body > p > img
1  <!DOCTYPE html>
2  <html lang="en">
3    <head> </head>
4    <body>
5      <h1>Tutorial AJAX sederhana</h1>
6      <p id="hint">Tutorial Fungsi Load()</p>
7      <p></p>
8    </body>
9  </html>
10

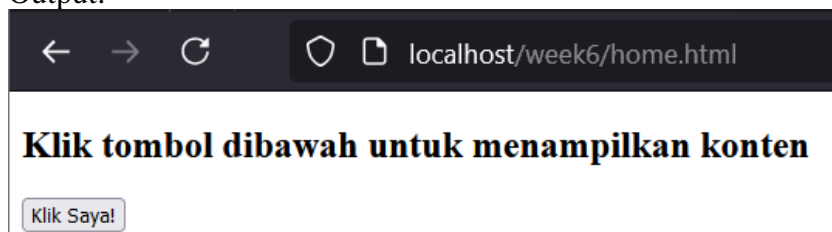
```

```

week6 > <> home.html > html > head > script > ready() callback > click() callback
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <script src="jquery-3.7.1.min.js"></script>
5      <script>
6        $(document).ready(function () {
7          $("button").click(function () {
8            $("#box").load("test-content.html");
9          });
10         });
11      </script>
12    </head>
13    <body>
14      <div id="box">
15        <h2>Klik tombol dibawah untuk menampilkan konten</h2>
16      </div>
17      <button type="button">Klik Saya!</button>
18    </body>
19  </html>
20

```

Output:





- The code uses jQuery's .load() function to dynamically load content from an external file (test-content.html) into a div when a button is clicked.
- Initially, the #box div contains static content, which is replaced by the loaded content after the button is clicked, all without refreshing the page.
- This is a simple and efficient example of using AJAX for dynamic content loading, improving the user experience by avoiding page reloads.

19. Reopen index.html file through the browser. Show the screenshot index.html and also provide the screenshot in step 5. Explain the difference with the result in step 5 (Question No. 17)

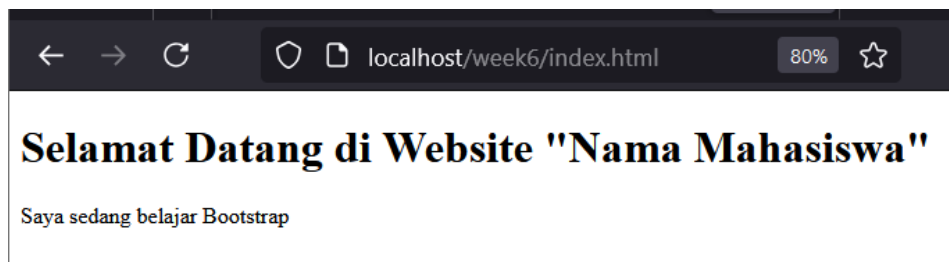
Code:

```

week6 > <> index.html > html > body > p
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6      <title>Pengenal Bootstrap</title>
7    </head>
8    <body>
9      <h1>Selamat Datang di Website "Nama Mahasiswa"</h1>
10     <p>Saya sedang belajar Bootstrap</p>
11   </body>
12 </html>
13

```

Output:



Code:

```
week6 > index.html > html > head > link
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6      <title>Pengenal Bootstrap</title>
7
8      <link
9        rel="stylesheet"
10       href="bootstrap-5.3.3-dist/bootstrap-5.3.3-dist/css/bootstrap.min.css"
11     />
12   </head>
13   <body>
14     <h1>Selamat Datang di Website "Nama Mahasiswa"</h1>
15     <p>Saya sedang belajar Bootstrap</p>
16   </body>
17 </html>
18
```

Output:



- Without Bootstrap: If the Bootstrap CSS is not found (due to an incorrect path or missing file), your HTML will display with default browser styles, leading to no formatting or responsive design features.
- With Bootstrap: When Bootstrap is properly linked (either locally or via CDN), your headings and paragraphs will inherit styles defined by Bootstrap, such as padding, font sizes, and responsive behavior.

20. Run container.html on a browser. Resize the browser to smaller, explain what happened and screenshot the result in the browser (Question No. 18)

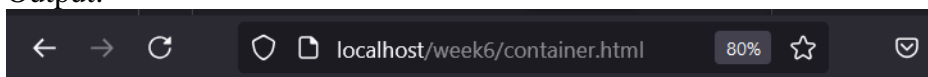
Code:

```

week6 > <> container.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6      <title>Pengenal Bootstrap</title>
7
8      <link
9        rel="stylesheet"
10       href="bootstrap-5.3.3-dist/bootstrap-5.3.3-dist/css/bootstrap.min.css"
11     />
12   </head>
13   <body>
14     <div class="container"></div>
15     <h1>Selamat Datang di Website "Nama Mahasiswa"</h1>
16     <p>Saya sedang belajar Bootstrap</p>
17   </body>
18 </html>
19

```

Output:



Selamat Datang di Website "Nama Mahasiswa"

Saya sedang belajar Bootstrap

The Bootstrap container ensures a responsive design by adjusting to different screen sizes. As the browser window shrinks, the container and its content adapt to provide an optimized view for smaller screens, maintaining readability and layout structure.

21. Run container.html on a browser. Change the size of the browser to smaller. Screenshot the result in your browser and explain the difference in step 3 (Question No. 19)

Code:

```

week6 > <> container.html > html > body > div.container-fluid
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6      <title>Pengenal Bootstrap</title>
7
8      <link
9        rel="stylesheet"
10       href="bootstrap-5.3.3-dist/bootstrap-5.3.3-dist/css/bootstrap.min.css"
11     />
12   </head>
13   <body>
14     <div class="container-fluid">
15       <h1>Selamat Datang di Website "Nama Mahasiswa"</h1>
16       <p>Saya sedang belajar Bootstrap</p>
17     </div>
18   </body>
19 </html>
20

```

Output:



Selamat Datang di Website "Nama Mahasiswa"

Saya sedang belajar Bootstrap

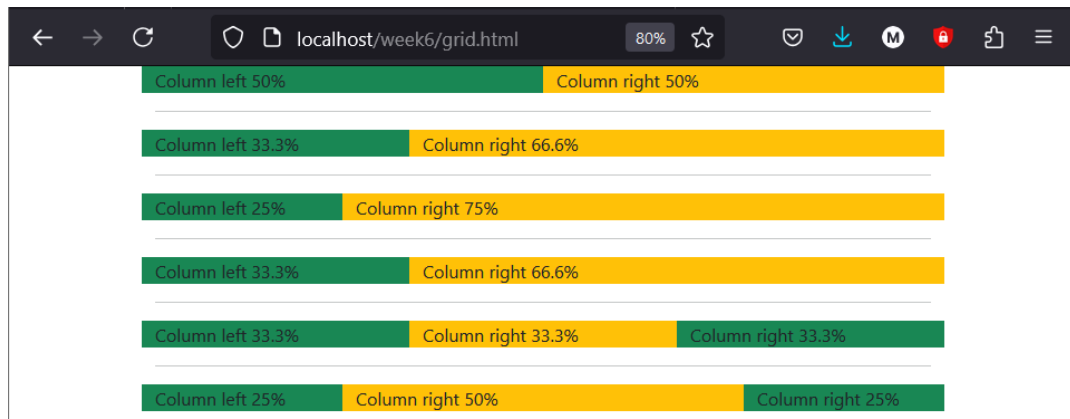
The main difference in this example is the use of container-fluid, which stretches the content across the entire width of the screen, while the standard container would have applied margins and restricted width, providing a more boxed layout.

22. Run grid.html on a browser. Explain what happened and screenshot the result in the browser (Question No. 20)

Code:

```
week6 > < grid.html > html > body > div.container
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6      <title>Pengelalan Bootstrap</title>
7
8      <link
9        rel="stylesheet"
10       href="bootstrap-5.3.3-dist/bootstrap-5.3.3-dist/css/bootstrap.min.css"
11     />
12   </head>
13   <body>
14     <div class="container">
15       <div class="row">
16         <div class="col-md-6 bg-success">Column left 50%</div>
17         <div class="col-md-6 bg-warning">Column right 50%</div>
18       </div>
19       <hr />
20       <div class="row">
21         <div class="col-md-4 bg-success">Column left 33.3%</div>
22         <div class="col-md-8 bg-warning">Column right 66.6%</div>
23       </div>
24       <hr />
25       <div class="row">
26         <div class="col-md-3 bg-success">Column left 25%</div>
27         <div class="col-md-9 bg-warning">Column right 75%</div>
28       </div>
29       <hr />
30       <div class="row">
31         <div class="col-md-4 bg-success">Column left 33.3%</div>
32         <div class="col-md-8 bg-warning">Column right 66.6%</div>
33       </div>
34       <hr />
35       <div class="row">
36         <div class="col-md-4 bg-success">Column left 33.3%</div>
37         <div class="col-md-4 bg-warning">Column right 33.3%</div>
38         <div class="col-md-4 bg-success">Column right 33.3%</div>
39       </div>
40       <hr />
41       <div class="row">
42         <div class="col-md-3 bg-success">Column left 25%</div>
43         <div class="col-md-6 bg-warning">Column right 50%</div>
44         <div class="col-md-3 bg-success">Column right 25%</div>
45       </div>
46     </div>
47   </body>
48 </html>
49
```

Output:



Each row has a different column layout, creating a variety of visual splits of the screen width. Bootstrap's grid system helps in making the layout responsive and organized without writing custom CSS for column widths.

23. Run grid2.html on a browser. Explain what happened and how it differed from step 3, and screenshot the result in the browser (Question No. 21)

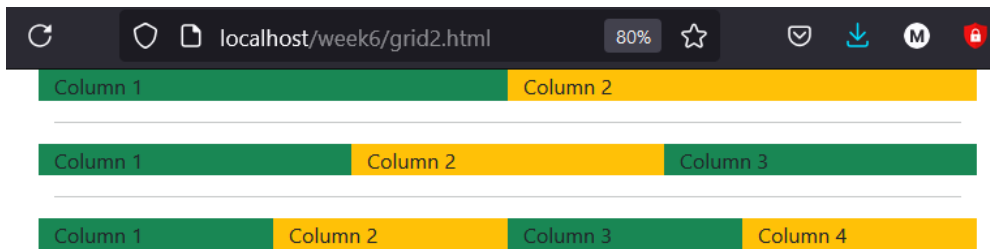
Code:

```

week6 > < grid2.html > html > body > div.container > div.row > div.col.bg-warning
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6      <title>Pengenalan Bootstrap</title>
7
8      <link
9        rel="stylesheet"
10       href="bootstrap-5.3.3-dist/bootstrap-5.3.3-dist/css/bootstrap.min.css"
11     />
12   </head>
13   <body>
14     <div class="container">
15       <div class="row">
16         <div class="col bg-success">Column 1</div>
17         <div class="col bg-warning">Column 2</div>
18       </div>
19       <hr />
20       <div class="row">
21         <div class="col bg-success">Column 1</div>
22         <div class="col bg-warning">Column 2</div>
23         <div class="col bg-success">Column 3</div>
24       </div>
25       <hr />
26       <div class="row">
27         <div class="col bg-success">Column 1</div>
28         <div class="col bg-warning">Column 2</div>
29         <div class="col bg-success">Column 3</div>
30         <div class="col bg-warning">Column 4</div>
31       </div>
32     </div>
33   </body>
34 </html>
35

```

Output:



- Column width distribution: Each row has a different number of columns. Since you're using just the col class without specifying sizes, Bootstrap automatically divides the row equally among the columns.
 - In the first row, 2 columns divide the space into 50% each.
 - In the second row, 3 columns divide the space into 33.3% each.
 - In the third row, 4 columns divide the space into 25% each.
- Background colors: The background color of each column alternates between green and yellow using Bootstrap's predefined background utility classes (bg-success for green and bg-warning for yellow).

24. Run grid3.html on a browser. Explain what happened and screenshot the result in the browser (Question No. 22)

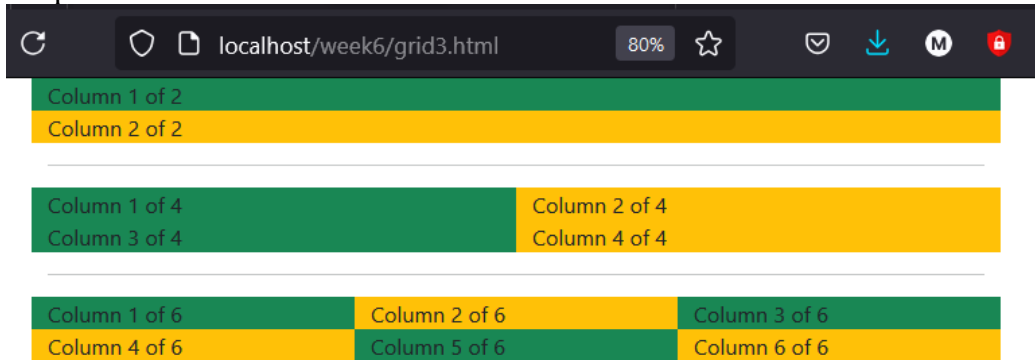
Code:

```

week6 > < grid3.html > < html > < body > < div.container > < div.row.row-cols-1
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6      <title>Pengenal Bootstrap</title>
7
8      <link
9        rel="stylesheet"
10       href="bootstrap-5.3.3-dist/bootstrap-5.3.3-dist/css/bootstrap.min.css"
11     />
12   </head>
13   <body>
14     <div class="container">
15       <div class="row row-cols-1">
16         <div class="col bg-success">Column 1 of 2</div>
17         <div class="col bg-warning">Column 2 of 2</div>
18       </div>
19       <hr />
20       <div class="row row-cols-2">
21         <div class="col bg-success">Column 1 of 4</div>
22         <div class="col bg-warning">Column 2 of 4</div>
23         <div class="col bg-success">Column 3 of 4</div>
24         <div class="col bg-warning">Column 4 of 4</div>
25       </div>
26       <hr />
27       <div class="row row-cols-3">
28         <div class="col bg-success">Column 1 of 6</div>
29         <div class="col bg-warning">Column 2 of 6</div>
30         <div class="col bg-success">Column 3 of 6</div>
31         <div class="col bg-warning">Column 4 of 6</div>
32         <div class="col bg-success">Column 5 of 6</div>
33         <div class="col bg-warning">Column 6 of 6</div>
34       </div>
35     </div>
36   </body>
37 </html>
38

```

Output:



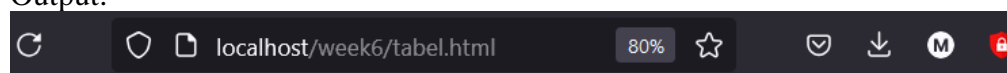
- Responsive Grid Layout: The HTML utilizes Bootstrap's grid system with row and col classes, allowing for a flexible and responsive design.
- Column Configuration:
 - First Row: Displays two columns (row-cols-1), stacked on small screens and side by side on larger screens.
 - Second Row: Contains four columns with row-cols-2, each occupying 50% of the width.
 - Third Row: Features six columns using row-cols-3, each taking about 33.3% of the row's width.
- Background Colors: The columns alternate between green (success) and yellow (warning) for visual distinction.
- Responsive Behavior: The columns stack vertically on smaller screens and adjust their widths according to the available space on larger screens.

25. Run tabel.html on a browser. Describe your observations and screenshot the results in your browser (Question No. 23)

Code:

```
week6 > tabel.html > html > body > div.container > table.table-bordered.table-striped.table-
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6      <title>Pengenan Bootstrap</title>
7
8      <link
9        rel="stylesheet"
10       href="bootstrap-5.3.3-dist/bootstrap-5.3.3-dist/css/bootstrap.min.css"
11     />
12   </head>
13   <body>
14     <div class="container">
15       <h2>Tabel Pelanggan</h2>
16       <table class="table table-bordered table-striped table-hover">
17         <tr>
18           <th>No.</th>
19           <th>Nama</th>
20           <th>Alamat</th>
21         </tr>
22         <tr>
23           <td>1.</td>
24           <td>Alifa</td>
25           <td>Malang</td>
26         </tr>
27         <tr>
28           <td>2.</td>
29           <td>Bima</td>
30           <td>Surabaya</td>
31         </tr>
32         <tr>
33           <td>3.</td>
34           <td>Cindy</td>
35           <td>Sidoarjo</td>
36         </tr>
37       </table>
38     </div>
39   </body>
40 </html>
41
```

Output:



Tabel Pelanggan

No.	Nama	Alamat
1.	Alifa	Malang
2.	Bima	Surabaya
3.	Cindy	Sidoarjo

- Table Structure: The page features a well-organized table with headers for "No.", "Nama," and "Alamat," along with three data rows.
- Bootstrap Styles:
 - Basic Styling: Bootstrap classes enhance the table's appearance.
 - Borders: The table-bordered class adds borders around the table and cells.
 - Striped Rows: The table-striped class alternates row colors for better readability.

- Hover Effects: The table-hover class highlights rows on mouse hover.
- Header Row: The header row stands out with bold text and a distinct background color.
- Responsiveness: The table adjusts its width based on the screen size, maintaining readability on various devices.
- Data Presentation: The information is presented clearly, making it easy to read and understand.

26. Run image.html on a browser. Change the size of the browser to smaller. Describe your observations and screenshot the results in your browser (Question No. 24)

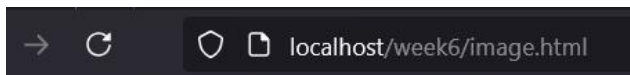
Code:

```

week6 > <> image.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6      <title>Pengenalan Bootstrap</title>
7
8      <link
9        rel="stylesheet"
10       href="bootstrap-5.3.3-dist/bootstrap-5.3.3-dist/css/bootstrap.min.css"
11     />
12    </head>
13    <body>
14      <div class="container">
15        <h2>Pengaturan Gambar pada Bootstrap</h2>
16        <div class="row row-cols-1">
17          <div class="col">
18            
19          </div>
20          <div class="col">
21            
22          </div>
23          <div class="col">
24            
25          </div>
26          <div class="col">
27            
28          </div>
29        </div>
30      </div>
31    </body>
32  </html>
33

```

Output:



Pengaturan Gambar pada Bootstrap



- **Image Display:** Four images are shown, each styled using Bootstrap classes:
 - **Rounded:** First image with rounded corners.
 - **Rounded Circle:** Second image displayed as a circle.
 - **Thumbnail:** Third image has a border and padding.
 - **Fluid:** Fourth image adjusts to fit the container while maintaining its aspect ratio.
- **Responsiveness:**
 - Images are arranged in a single row in larger viewports but shift to a single column in smaller sizes.
 - The img-fluid class ensures images resize appropriately without overflow.
- **Visual Clarity:** Distinct styles for each image remain clear even when stacked vertically.
- **Bootstrap Grid:** The layout adapts seamlessly with the responsive grid system provided by Bootstrap.

27. Run jumbotron.html on your browser. Describe your observations and screenshot the results in your browser (Question No. 25)

Code:

```
week6 > <j> jumbotron.html > html > body > div.container > p
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6      <title>Pengenal Bootstrap</title>
7
8      <link
9        rel="stylesheet"
10       href="bootstrap-5.3.3-dist/bootstrap-5.3.3-dist/css/bootstrap.min.css"
11     />
12   </head>
13   <body>
14     <div class="container">
15       <div class="jumbotron">
16         <h1>Belajar Bootstrap</h1>
17         <p>
18           Bootstrap adalah framework untuk membuat responsif website menggunakan
19           HTML, CSS, and JS.
20         </p>
21       </div>
22       <p>Mata Kuliah Desain dan Pemrograman web</p>
23       <p>Nama Mahasiswa - NIM</p>
24     </div>
25   </body>
26 </html>
27
```

Output:

Belajar Bootstrap

Bootstrap adalah framework untuk membuat responsif website menggunakan HTML, CSS, and JS.

Mata Kuliah Desain dan Pemrograman web

Nama Mahasiswa - NIM

- Jumbotron: This large, attention-grabbing area is useful for titles or introductory text, but note that in Bootstrap 5, the jumbotron class has been removed. You would now use a combination of utilities such as bg-light or custom classes for similar effects.
- The page is basic but well-structured, utilizing Bootstrap to organize and present the content in a clean, responsive manner.

28. Run alert.html on your browser. Describe your observations and screenshot the results in your browser (Question No. 26)

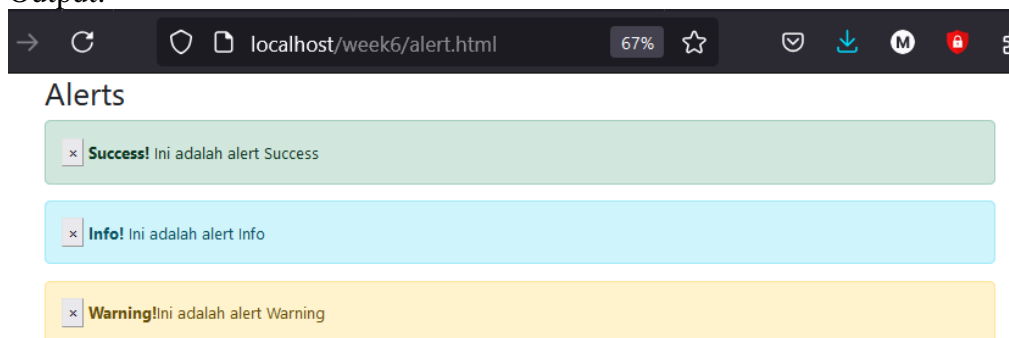
Code:

```

week6 > alert.html > html > body > div.container > div.alert.alert-warning.alert-dismissible
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6      <title>Pengenal Bootstrap</title>
7
8      <link
9        rel="stylesheet"
10       href="bootstrap-5.3.3-dist/bootstrap-5.3.3-dist/css/bootstrap.min.css"
11     />
12   </head>
13   <body>
14     <div class="container">
15       <h2>Alerts</h2>
16       <div class="alert alert-success alert-dismissible">
17         <button type="button" class="close" data-dismiss="alert">
18           &times;
19         </button>
20         <strong>Success!</strong> Ini adalah alert Success
21       </div>
22       <div class="alert alert-info alert-dismissible">
23         <button type="button" class="close" data-dismiss="alert">
24           &times;
25         </button>
26         <strong>Info!</strong> Ini adalah alert Info
27       </div>
28       <div class="alert alert-warning alert-dismissible">
29         <button type="button" class="close" data-dismiss="alert">
30           &times;
31         </button>
32         <strong>Warning!</strong>Ini adalah alert Warning
33       </div>
34     </div>
35   </body>
36 </html>
37

```

Output:



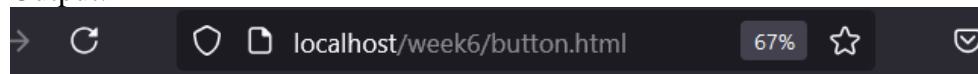
- Alert Boxes: Three different types of Bootstrap alerts are displayed:
 - Success Alert: A green alert box with a "Success!" message.
 - Info Alert: A blue alert box with an "Info!" message.
 - Warning Alert: A yellow alert box with a "Warning!" message.
- Dismissible Functionality: Each alert has a close button (styled with an × symbol). Clicking the close button removes the alert from the page.
- Visual Styling: The alerts are styled using Bootstrap's alert-success, alert-info, and alert-warning classes, which provide background colors and padding for visual clarity.
- Responsive Behavior: The layout adapts to different screen sizes. Alerts remain visually distinct and adjust well even on smaller screens.

29. Run button.html on a browser. Describe your observations and screenshot the results in your browser (Question No. 27)

Code:

```
week6 > <> button.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6      <title>Pengelalan Bootstrap</title>
7
8      <link
9        rel="stylesheet"
10       href="bootstrap-5.3.3-dist/bootstrap-5.3.3-dist/css/bootstrap.min.css"
11     />
12   </head>
13   <body>
14     <div class="container">
15       <h2>Button Styles</h2>
16       <button type="button" class="btn">Basic</button>
17       <button type="button" class="btn btn-primary">Primary</button>
18       <button type="button" class="btn btn-secondary">Secondary</button>
19       <button type="button" class="btn btn-success">Success</button>
20       <button type="button" class="btn btn-info">Info</button>
21       <button type="button" class="btn btn-warning">Warning</button>
22       <button type="button" class="btn btn-danger">Danger</button>
23       <button type="button" class="btn btn-dark">Dark</button>
24       <button type="button" class="btn btn-light">Light</button>
25       <button type="button" class="btn btn-link">Link</button>
26     </div>
27   </body>
28 </html>
29
```

Output:



Button Styles

Basic Primary Secondary Success Info Warning Danger Dark Light [Link](#)

- Each button is created using the btn class followed by a specific modifier class (e.g., btn-primary, btn-success) to apply different colors and styles.
- These button classes are part of Bootstrap's core components, designed to provide a consistent look and feel across different types of buttons.
- The colors and styles follow conventional UI patterns (e.g., green for success, red for danger), making the buttons intuitive to use.

30. Run navbar.html on a browser. Describe your observations and screenshot the results in the browser (Question No. 28)

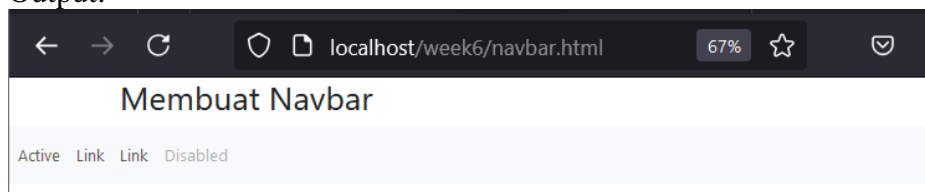
Code:


```

week6 > <> navbar.html > html > body > nav.navbar.navbar-expand-sm.bg-light.navbar-light
1  <!DOCTYPE html>
2  <html lang="en">
3
4    <head>
5      <meta charset="UTF-8" />
6      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7      <title>Pengenal Bootstrap</title>
8
9      <link
10       rel="stylesheet"
11       href="bootstrap-5.3.3-dist/bootstrap-5.3.3-dist/css/bootstrap.min.css"
12     />
13   </head>
14   <body>
15     <div class="container">
16       <h2>Membuat Navbar</h2>
17     </div>
18     <nav class="navbar navbar-expand-sm bg-light navbar-light">
19       <ul class="navbar-nav">
20         <li class="nav-item active">
21           <a class="nav-link" href="#">Active</a>
22         </li>
23         <li class="nav-item">
24           <a class="nav-link" href="#">Link</a>
25         </li>
26         <li class="nav-item">
27           <a class="nav-link" href="#">Link</a>
28         </li>
29         <li class="nav-item">
30           <a class="nav-link disabled" href="#">Disabled</a>
31         </li>
32       </ul>
33     </nav>
34   </body>
35 </html>

```

Output:



When this code is executed (with the typo corrected), you will see a navbar with the following items:

- An active link that stands out visually (representing the current page).
- Two normal links for navigation.
- One disabled link that appears grayed out and unclickable

The entire navbar will be light-colored due to the bg-light class, and the items will follow Bootstrap's responsive behavior, ensuring the layout adjusts well on different screen sizes.

31. Run navbar2.html on a browser. Describe your observations and screenshot the results in your browser (Question No. 29)

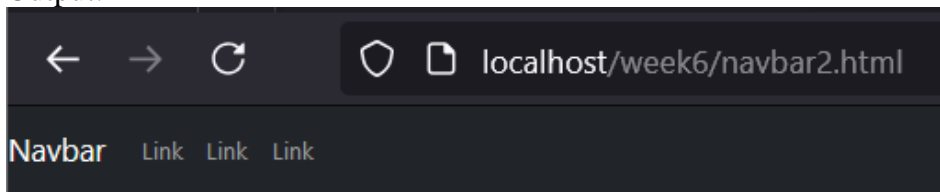
Code:

```

week6 > navbar2.html > html
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8" />
5 <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6 <title>Pengenalan Bootstrap</title>
7
8 <link
9   rel="stylesheet"
10  href="bootstrap-5.3.3-dist/bootstrap-5.3.3-dist/css/bootstrap.min.css"
11 />
12 </head>
13 <body>
14 <nav class="navbar navbar-expand-md bg-dark navbar-dark">
15   <a class="navbar-brand" href="#">Navbar</a>
16   <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#collapsibleNavbar">
17     <span class="navbar-toggler-icon"></span>
18   </button>
19   <div class="collapse navbar-collapse" id="collapsibleNavbar">
20     <ul class="navbar-nav">
21       <li class="nav-item">
22         <a class="nav-link" href="#">Link</a>
23       </li>
24       <li class="nav-item">
25         <a class="nav-link" href="#">Link</a>
26       </li>
27       <li class="nav-item">
28         <a class="nav-link" href="#">Link</a>
29       </li>
30     </ul>
31   </div>
32 </nav>
33 </body>
34 </html>
35

```

Output:



- On medium and larger screen sizes, the navbar will display horizontally with the "Navbar" brand on the left and the links aligned next to it.
- On smaller screens, the links will collapse into a toggleable section that is hidden by default. You will see the "hamburger" icon (the three horizontal lines), which allows users to click and expand the links.
- The navbar uses a dark background with light text because of the bg-dark and navbar-dark classes.