Modern Mondrian

1. Make a new folder called **ModernMondrian**.
2. In the **ModernMondrian** folder, create a **css** folder.
3. In **Brackets**, open the **ModernMondrian** folder as a project.
4. Create a **new file** in the root of the **ModernMondrian** folder and name it **index.html**.
5. Add in the base **HTML5** structure.
6. In the **<head>** tag, add a <link> element with the following:
    - an href equal to: "css/index.css"
    - a rel attribute with a value of "stylesheet"
    - and the optional type with a value of "text/css"

7. Between the **<body>** tags, add the following code:

   <div id="painting"></div>

8. Between the **<div id="painting">** tags, add the following code:

   ```
   <body>
      <div id="painting">
        <div id="toprow">
            <div id="bigbox" class="red"></div>
        </div>
      </div>
   </body>
   ```

9. Save the index.css file, then create a **new file** in the **css** folder and name it i**ndex.css**.
10. In **index.css**, add the following code to style the html:

    ```
    body, html {
       background-color: #f6f6f6;
    }

    #painting {
       background-color: #fff4db;
       width: /* To calculate the width, refer to the mondrian_grid.jpg file for the dimensions */;
       height: /* To calculate the height, refer to the mondrian_grid.jpg file for the dimensions */;
       margin-top: 30px;
       margin-left: auto;
       margin-right: auto;
       box-shadow: 10px 10px 12px #8d8d8d;
    }

    #toprow {
       height: 290px;
    }
    ```

```
#bigbox {
   width: 290px;
   height: 290px;
}
```

11. In **index.css**, create the following code for the colours:

- Create a class named "**red**", with a **background colour** property of "**red**"
- Create a class named "**blue**", with a **background colour** property of "**blue**"
- Create a class named "**yellow**", with a **background colour** property of "**yellow**"
- Create a class named "**black**", with a **background colour** property of "**black**"

12. **Save** the files and run a **preview**.

13. In **index.css**, add the following code:

- Add a **class** of "**right**" with a **float** property of **right**

14. In **index.html**, add '**right**' to the class of the **<div>** with **id="bigbox"**.
15. In **index.css**, add the following code:

```
#divider1 {
   height: 290px;
   width: 10px;
}
```

16. In index.html, add the following:

```
<body>
   <div id="painting">
      <div id="toprow">
         <div id="bigbox" class="red right"></div>
         <div id="divider1" class="black right"></div>
      </div>
   </div>
</body>
```

17. Now let's make a column, called #topleftcolumn. We will be putting three boxes in it, two with the .mediumbox class, and a horizontal #divider2 in between.

    #topleftcolumn won't be visible, but it helps position and stack those three boxes vertically within that last bit of the row.

18. In **index.html**, add the following:

```
<body>
   <div id="painting">
      <div id="toprow">
```

```
        <div id="bigbox" class="red right"></div>
        <div id="divider1" class="black right"></div>
        <div id="topleftcolumn" class= "right">
            <div class="mediumbox"></div>
            <div id="divider2" class= "black"></div>
            <div class="mediumbox"></div>
        </div>
      </div>
    </div>
</body>
```

19. In **index.css**, add the following:

```
.mediumbox {
    width: 100px;
    height: 140px;
}

#divider2 {
    height: 10px;
    width: 100px;
}
```

20. **Save** and run a **preview**.
21. In **index.html**, add the following:

```
<body>
    <div id="painting">
        <div id="toprow">
            <div id="bigbox" class="red right"></div>
            <div id="divider1" class="black right"></div>
            <div id="topleftcolumn" class= "right">
                <div class="mediumbox"></div>
                <div id="divider2" class= "black"></div>
                <div class="mediumbox"></div>
            </div>
        </div>
        <div id="middlerow" class="black"></div>
    </div>
</body>
```

22. In **index.css**, add the following code:

```
#middlerow {
    height: 10px;
}
```

23. **Save** and run a **preview**.

24. In **index.html**, add the following code:

```html
<body>
    <div id="painting">
        <div id="toprow">
            <div id="bigbox" class="red right"></div>
            <div id="divider1" class="black right"></div>
            <div id="topleftcolumn" class= "right">
                <div class="mediumbox"></div>
                <div id="divider2" class= "black"></div>
                <div class="mediumbox"></div>
            </div>
        </div>
        <div id="middlerow" class="black"></div>
        <div id= "bottomrow">
            <div id= "bottomrightcolumn" class= "right">
                <div id= "tinybox"></div>
                <div id= "divider4" class= "black"></div>
                <div id= "tinybox" class= "yellow"></div>
            </div>
            <div id= "divider3" class= "black right"> </div>
            <div id= "widebox" class= "right"> </div>
            <div id= "divider3" class= "black right"> </div>
            <div id= "smallbox" class= "blue right"> </div>
        </div>
    </div>
</body>
```

25. In **index.css**, add the following code:

```css
#smallbox {
   width: 100px;
   height: 100px;
}

#widebox {
   width: 260px;
   height: 100px;
}

#tinybox {
   width: 20px;
   height: 45px;
}

#divider3 {
   height: 100px;
   width: 10px;
```

```
}

#divider4 {
    height: 10px;
    width: 20px;
}
```

26. **Save** and run a **preview**.

So far we've been dealing with pixel dimensions on everything. What happens if we change height and width to percents? Can you create a Mondrian-style image that fills the entire browser and scales to any dimension the window takes?