**VR Project**

**ABSTRACT:**

In this project “VR - home”, user will be able to understand the specifications of the product “Oculus”.

**FOLDER STRUCTURE OF MULTILINGUAL GREETINGS:**

Index.html 🡪 HTML file

Style.css 🡪 Stylesheet/CSS file

Source 🡪 Images container

**STEP TO CREATE A FOLDER:**

Step 1: Create a folder named “VR Project” or the name you prefer. But it is recommended to use the folder name and the project name as same.

Step 2: Create the normal text document file and change its extension to “.html” along with name “index”, which represents the webpage of our project.

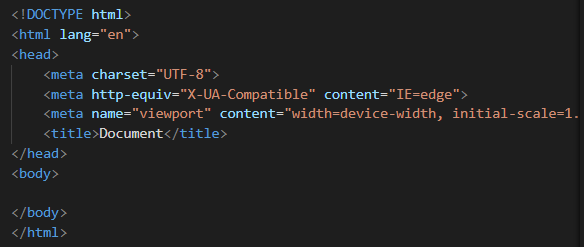
Step 3: Now, create a normal text document like the previous one and change it’s extension to “.css”, which is used to style our webpage.

Step 4: Create a folder and store all the required for our project.

**index.html:**

this “index.html” is our base file, which helps us to provide content for our webpage.

The basic structure of an HTML file is shown below.



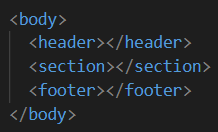
This structure can be simply brought into our file by pressing a shortcut key shift+1(one). This will show an exclamatory(!) symbol. If you press enter, then it will provide the base structure code like shown in the image.

Change the title into FAQs which will be provided in the title tag of the html file.

Add link tag to link our external CSS file to our html file.



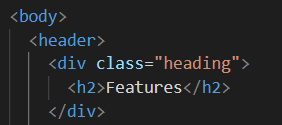
The body section of our html will be containing a basic structure like shown below.



The given code is a snippet of HTML markup that defines the beginning of a web page's body section. Within the body section, there is a header element that contains two nested div elements.

The first div element has a class attribute set to "heading", which is likely used for styling purposes. Within this div element, there is an h2 element that displays the text "Features". This is likely a heading that introduces a section of the page that highlights the features of a product or service.

The second div element within the header section has no class or id attributes, but it may be used for additional styling or layout purposes. This div contains no content of its own and appears to serve as a container for the h2 element.



The given code is a snippet of HTML markup that defines a section of a web page dedicated to showcasing the features of a product or service. The features are presented in a grid-like format, with each feature represented by a "box" element. These boxes are contained within a div element with the id attribute set to "features". The features include the following:

"Easy To connect": This feature is represented by a box element with the class attribute set to "box item-1". The box contains an image of an easy-connect icon and a short description of the feature's benefits.

"VR capability": This feature is represented by a box element with the class attribute set to "box item-2". The box contains an image of a VR headset and a short description of the feature's benefits.

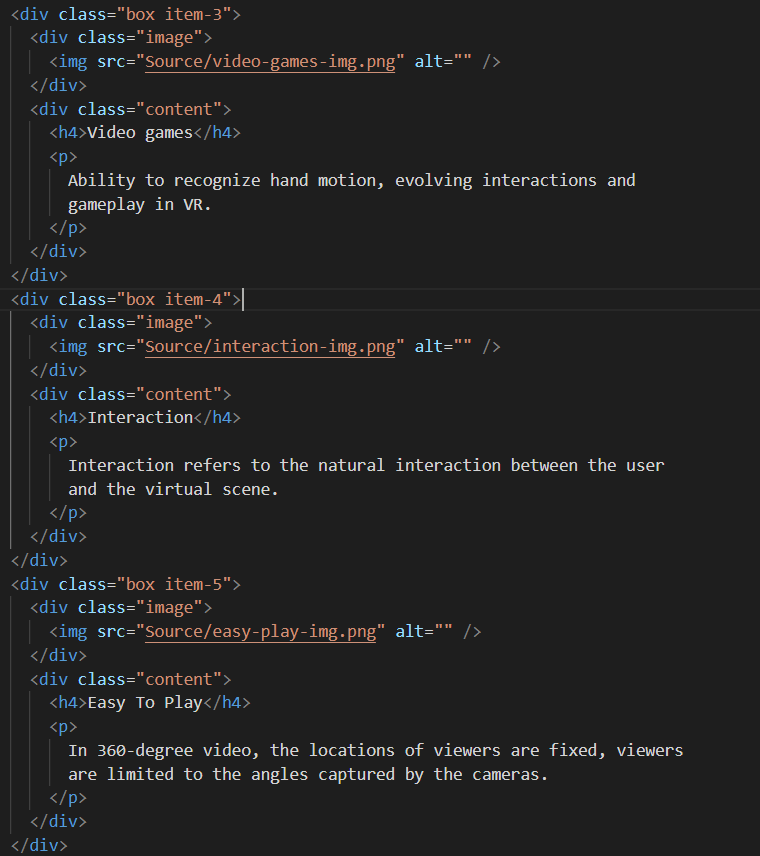
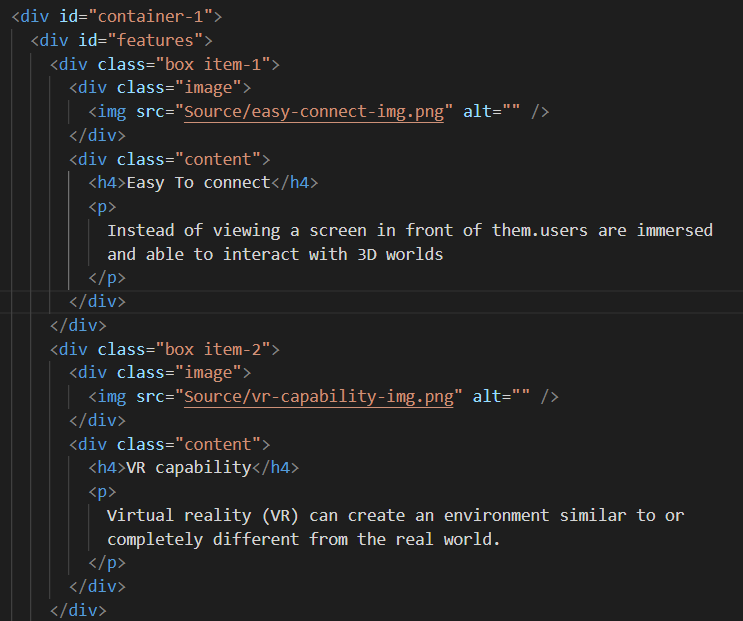
"Video games": This feature is represented by a box element with the class attribute set to "box item-3". The box contains an image of a video game controller and a short description of the feature's benefits.

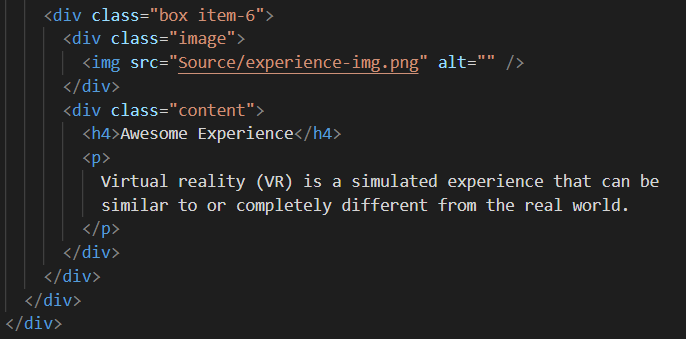
"Interaction": This feature is represented by a box element with the class attribute set to "box item-4". The box contains an image of a hand gesture and a short description of the feature's benefits.

"Easy To Play": This feature is represented by a box element with the class attribute set to "box item-5". The box contains an image of a play button and a short description of the feature's benefits.

"Awesome Experience": This feature is represented by a box element with the class attribute set to "box item-6". The box contains an image of an experience icon and a short description of the feature's benefits.

Each box element consists of two nested div elements: one with the class attribute set to "image" and the other with the class attribute set to "content". The "image" div contains an img element that displays an image related to the feature, while the "content" div contains an h4 element that displays the feature's name and a p element that displays a short description of the feature. Overall, the code creates a visually appealing and informative feature section for a product or service.

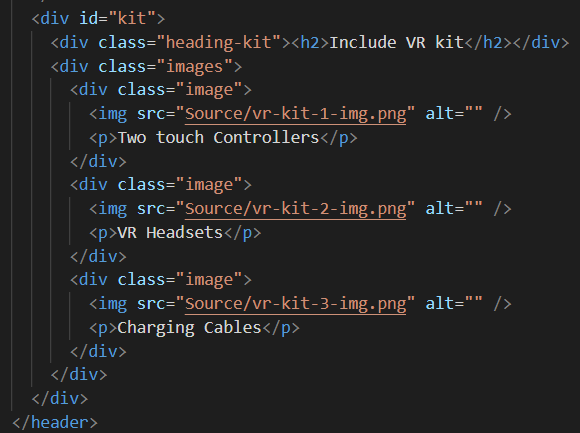




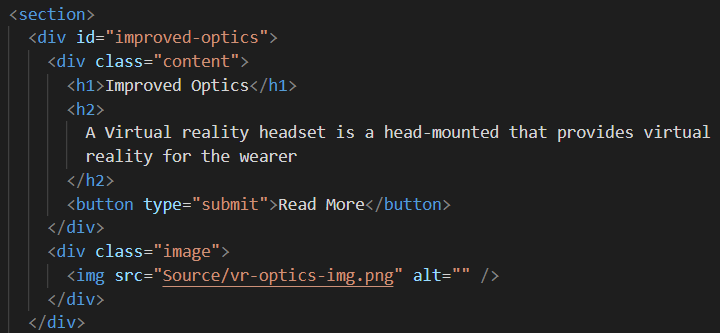
This code represents a section with the id "kit" that contains information about the VR kit that is required to experience the VR content. The section includes a heading with the class "heading-kit" and an h2 element that displays the text "Include VR kit".

Below the heading, there is a div element with the class "images" that contains three separate divs with the class "image". Each of these divs includes an image and a caption describing the item in the image.

The first "image" div includes an image with the source "Source/vr-kit-1-img.png" and a caption that reads "Two touch Controllers". The second "image" div includes an image with the source "Source/vr-kit-2-img.png" and a caption that reads "VR Headsets". The third "image" div includes an image with the source "Source/vr-kit-3-img.png" and a caption that reads "Charging Cables".



Overall, this code represents a section that explains what is included in the VR kit needed to experience the VR content, which consists of two touch controllers, VR headsets, and charging cables.

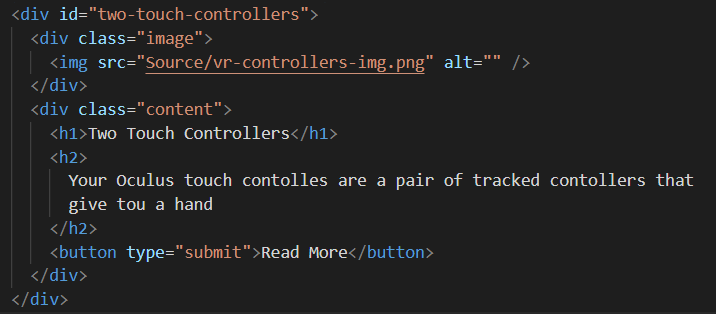


This code represents a section with the id "improved-optics". It consists of two div elements, one with the class "content" and the other with the class "image".

Inside the "content" div, there is an h1 element with the text "Improved Optics", an h2 element with a description of what a Virtual reality headset is, and a button with the text "Read More".

In the "image" div, there is an img element with the source "Source/vr-optics-img.png" and an empty alt attribute.

Overall, this section seems to be introducing the concept of virtual reality headsets and highlighting the improved optics that they provide. The "Read More" button suggests that there is more information available on this topic.



This code represents a section of a webpage that describes the "Two Touch Controllers" component of a virtual reality system. The section consists of a div with an id of "two-touch-controllers" that contains two child divs - one with a class of "image" and another with a class of "content".

The "image" div contains an image tag with a source file of "Source/vr-controllers-img.png" and an empty alt attribute. The image shows an illustration of the two hand-held touch controllers.

The "content" div contains two heading tags - h1 and h2 - and a button tag. The h1 tag displays the title "Two Touch Controllers" and the h2 tag provides additional information about the controllers. The button tag allows the user to navigate to another page with more information about the controllers when clicked.

Overall, this section provides a brief introduction to the "Two Touch Controllers" component of the VR system, and invites the user to learn more about it.

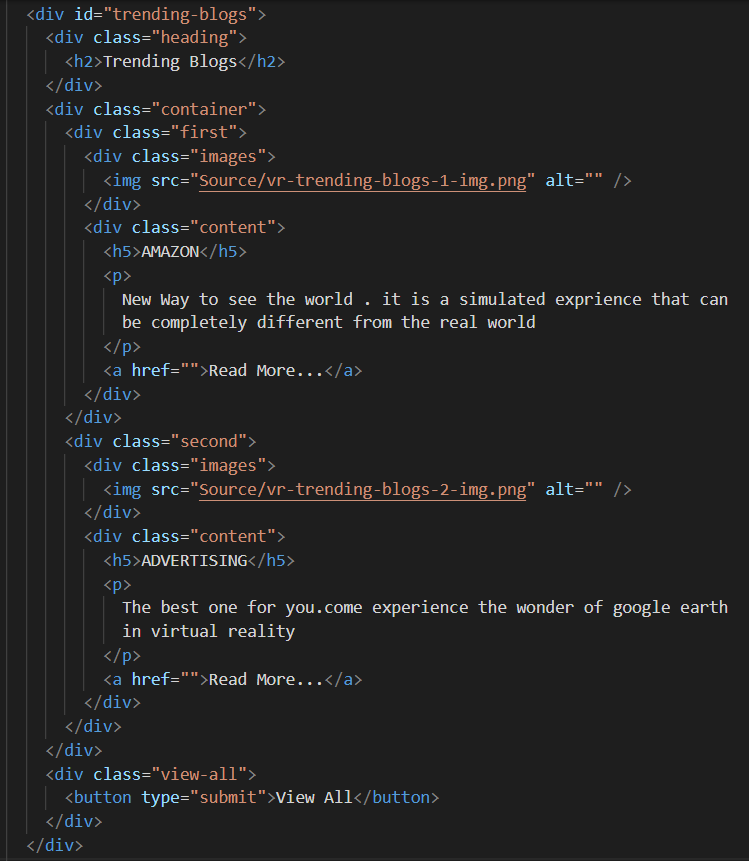


This code represents a banner section on a webpage, which includes an image and a content area. The div element with an id of banner wraps around the entire banner section. Inside the div, there are two child elements: div with a class of image and div with a class of content.

The image div contains an img element with a source file path to an image named vr-set-img.png which is an illustration of a VR headset.

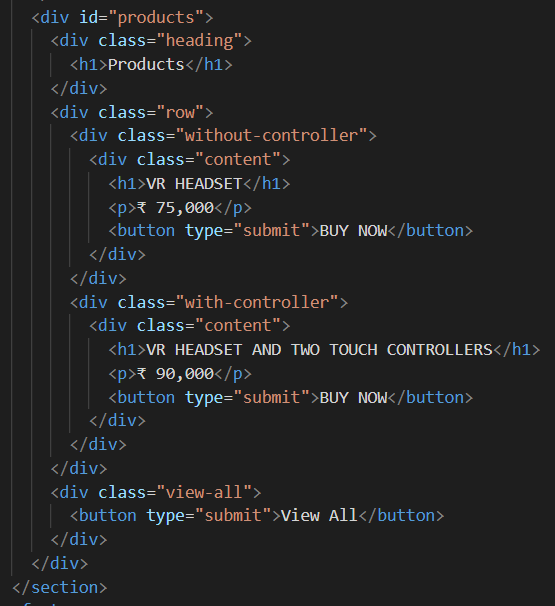
The content div contains an h1 element with the text "Virtually Anything is Possible" and a button element with the text "Contact Us".

This code can be used as a banner section for a VR company or a webpage that promotes VR products or services. The banner section displays an image of a VR headset and invites users to take action by clicking the "Contact Us" button.



This code represents a section with a heading "Trending Blogs" which contains two blog posts. The section has a container with two divs named "first" and "second" and each div has an image and a content section. The content section contains a heading, short description, and a link to read more. At the bottom of the section, there is a button to view all blog posts.

The HTML code consists of multiple nested div elements with class and id attributes. The div with id "trending-blogs" contains a div with a class "heading" which has an h2 heading element. The div with class "container" contains two child divs with class "first" and "second". Each of these divs contains a div with class "images" which has an image element and a div with class "content" which has an h5 heading element, a p element, and an anchor element. The last div with class "view-all" contains a button element.

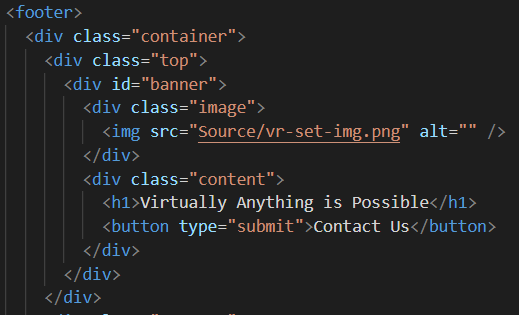


This code defines a section with the id "products". Inside the section, there is a heading with the text "Products".

The section also contains a row with two product options, each enclosed within a div with a class "without-controller" and "with-controller" respectively.

Each product option has a heading with the product name, a price, and a "Buy Now" button. The first product option includes only the VR headset, and it costs ₹ 75,000. The second product option includes both the VR headset and two touch controllers and costs ₹ 90,000.

Finally, there is a "View All" button that allows the user to see more product options or related products.

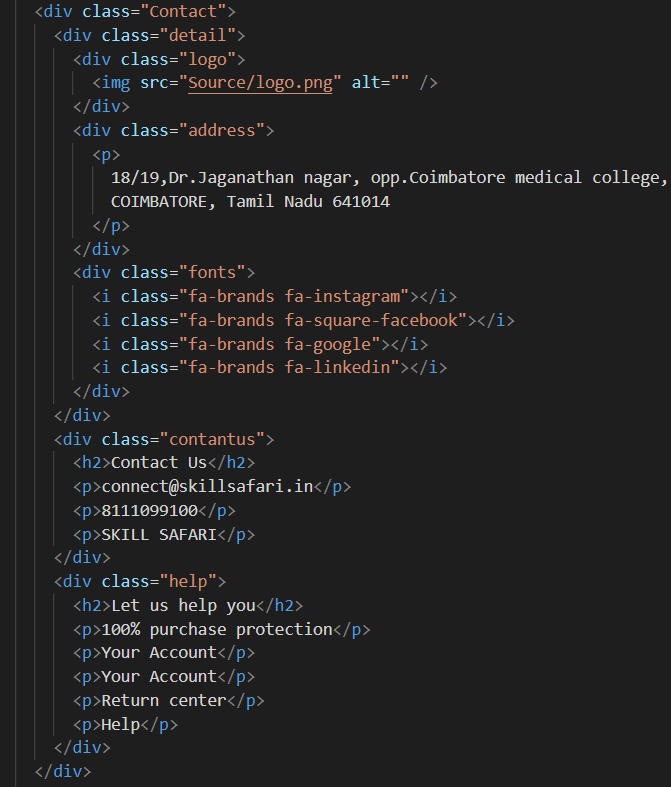


This is a HTML code for a footer section of a website. It contains a container with two main divs - top and bottom.

The top div contains a banner with an image of a VR set and a heading "Virtually Anything is Possible" along with a button to contact the company.

The bottom div may contain other information such as links to social media pages, terms and conditions, privacy policy, etc. which are not shown in this code snippet.

Overall, the code is a basic structure of a footer section for a website.

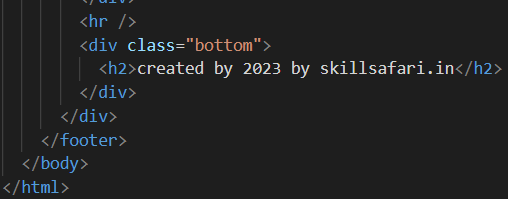


This code defines a section of HTML code with the class "Contact" that contains information related to contacting the website or company. The section is divided into three parts:

The first part contains the company logo, address, and social media icons for Instagram, Facebook, Google, and LinkedIn.

The second part contains the heading "Contact Us" and contact details such as email, phone number, and company name.

The third part contains the heading "Let us help you" and a list of options related to account management and returns, as well as a general help option.



This code is the end of an HTML document and it contains the closing tags for the <body> and <html> elements. It also includes a horizontal line using the <hr> element to visually separate the content above it from the footer.

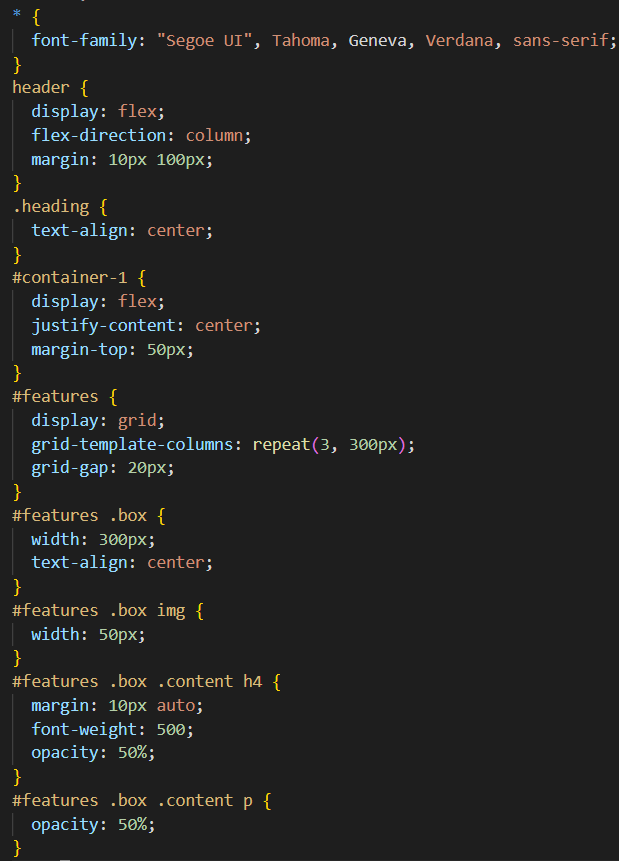
The <footer> element contains information about the website creator and the copyright date. It has three <div> elements with class names: "Contact", "bottom", and "container".

The "Contact" <div> contains details about contacting the website creators, including their logo, address, and social media links.

The "bottom" <div> contains a statement about who created the website and when.

Overall, this code represents the footer section of an HTML document that provides information about the website creators and their contact details.

**CSS file:**



This is CSS code that sets styles for various elements on a webpage. Here's a brief explanation of what each section does:

The \* selector sets a default font for all elements on the page.

The header selector sets the display to flex and the flex-direction to column, which will stack the child elements vertically. It also sets a margin of 10px on the top and bottom and 100px on the left and right.

The .heading class sets the text-align to center.

The #container-1 selector sets the display to flex and the justify-content to center, which will horizontally center its child elements. It also sets a margin-top of 50px.

The #features selector sets the display to grid and the grid-template-columns to repeat 3 times with a width of 300px each. It also sets a grid-gap of 20px between the grid items.

The #features .box selector sets the width of each box to 300px and centers the text within each box.

The #features .box img selector sets the width of the image within each box to 50px.

The #features .box .content h4 selector sets a margin of 10px on the top and bottom, sets the font weight to 500, and reduces the opacity to 50%.

The #features .box .content p selector reduces the opacity of the text to 50%.



“#kit” is a selector that targets an element with the ID "kit".

text-align: center centers the text within the element horizontally.

.images is a selector that targets elements with the class "images".

display: flex sets the display property of the element to a flexible container, allowing its child elements to be laid out in a row or a column.

justify-content: center centers the child elements along the horizontal axis.

#kit .image is a selector that targets an element with the class "image" that is a descendant of the element with the ID "kit".

width: 300px and height: 300px set the dimensions of the element to 300 pixels in width and height.

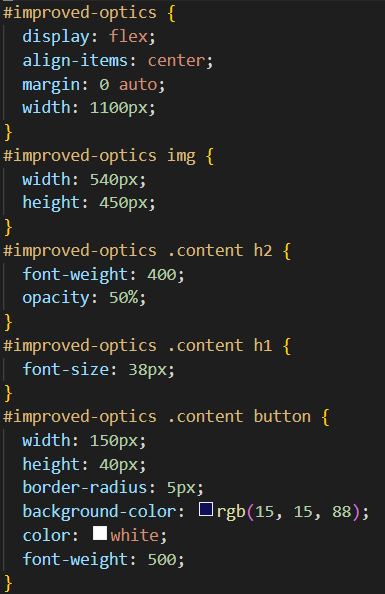
margin: 20px adds a margin of 20 pixels around the element.

#kit .images img is a selector that targets an image element that is a descendant of an element with the class "images" that is a descendant of an element with the ID "kit".

width: 100% and height: 300px sets the dimensions of the image element to 100% width and 300 pixels height.

section is a selector that targets all section elements on the page.

margin-top: 50px adds a margin of 50 pixels at the top of each section.



This code defines styles for a section with id "improved-optics" that includes an image and some content. Here is a breakdown of the CSS:

#improved-optics: This selects the element with the id "improved-optics".

display: flex: This sets the display property of the element to flex, allowing its child elements to be laid out in a flexible row or column.

align-items: center: This centers the child elements of the flex container along the vertical axis.

margin: 0 auto: This centers the element horizontally by setting the left and right margins to "auto".

width: 1100px: This sets the width of the element to 1100 pixels.

img: This selects any images within the element.

width: 540px; height: 450px: This sets the width and height of the image to 540 pixels and 450 pixels, respectively.

.content h2: This selects any h2 elements within a child element with the class "content".

font-weight: 400: This sets the font weight of the h2 element to 400.

opacity: 50%: This sets the opacity of the h2 element to 50%.

.content h1: This selects any h1 elements within a child element with the class "content".

font-size: 38px: This sets the font size of the h1 element to 38 pixels.

.content button: This selects any buttons within a child element with the class "content".

width: 150px: This sets the width of the button to 150 pixels.

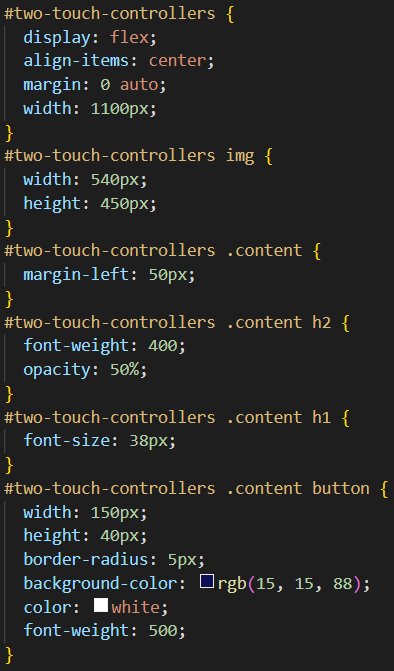
height: 40px: This sets the height of the button to 40 pixels.

border-radius: 5px: This sets the border radius of the button to 5 pixels, giving it rounded corners.

background-color: rgb(15, 15, 88): This sets the background color of the button to a dark blue.

color: white: This sets the text color of the button to white.

font-weight: 500: This sets the font weight of the button to 500.

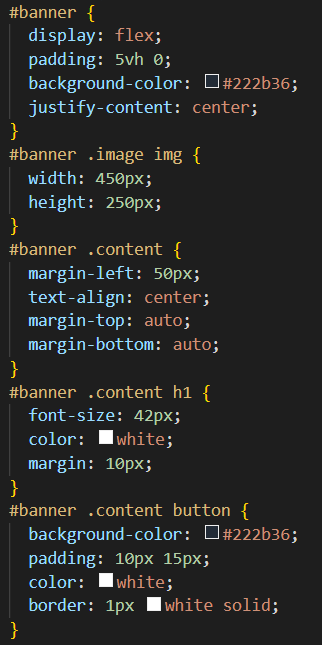


This is CSS code that styles a section with the ID "two-touch-controllers". It sets the section to display as a flex container and aligns its items vertically to the center. It also sets the left and right margins to auto and the width to 1100px.

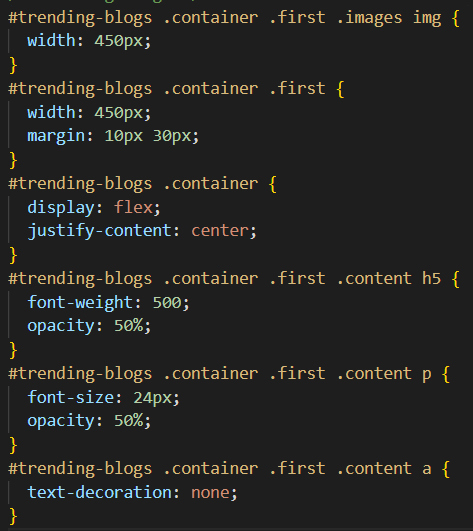
Inside the section, there is an image with the ID "two-touch-controllers img" that has a width of 540px and a height of 450px.

There is also a div with the class "content" that contains two headings, one with an h2 tag and another with an h1 tag. The h2 tag has a font-weight of 400 and an opacity of 50%, while the h1 tag has a font-size of 38px.

Finally, there is a button inside the "content" div with a width of 150px, a height of 40px, and a border-radius of 5px. It has a background color of rgb(15, 15, 88), a text color of white, and a font-weight of 500.



This code defines the styling for a banner section in a web page. The banner section has a dark background color (#222b36) and is centered on the page using the "justify-content: center" property. The banner is made up of an image and some content, which are displayed side-by-side using flexbox. The image is given a fixed width of 450px and a height of 250px using the "width" and "height" properties. The content section is given a margin on the left to create some space between the image and the text.



This code defines the styles for a section of a web page called #trending-blogs. The #trending-blogs section is designed to display a set of trending blogs. The section has a container with class .container, which is set to flex and centered horizontally with justify-content: center;.

The first blog in the container has a class .first and is set to a fixed width of 450px with a margin of 10px on top and bottom and 30px on the left and right. The blog content has a heading with a font-weight of 500 and an opacity of 50%. The paragraph text has a font-size of 24px and an opacity of 50%. The link within the content has no text-decoration.

The blog's image has a width of 450px, while the images of other blogs in the container are not defined in this code. Therefore, they will inherit the default value or take up the available space based on their dimensions.

Overall, this code defines the layout and styles of the first blog in the trending blogs section. The use of a fixed width and margin creates consistent spacing between the blogs, while the opacity reduces the contrast between the text and background for a subtle effect.



This code is for styling a section of a web page called "trending-blogs". The section contains two blogs and a "view all" button. Let's go through the CSS code line by line to understand what each section does.

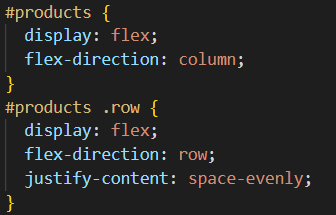
The first block of code styles the first blog in the section. The image inside the blog is given a width of 450px to fit the blog container. The blog container itself is given a width of 450px and a margin of 10px on the top and bottom and 30px on the left and right.

The second block of code styles the second blog in the section. The image inside the blog is also given a width of 450px. The blog container is given a width of 450px and a height of 420px, which is larger than the first blog. It also has the same margin as the first blog.

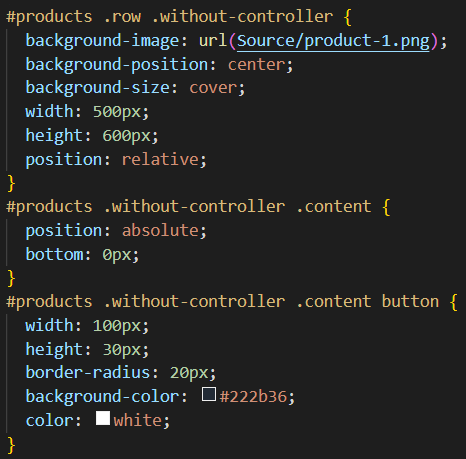
The next three blocks of code style the content inside the blogs. Both blogs have a heading with a font weight of 500 and an opacity of 50%. They also have a paragraph with a font size of 24px and an opacity of 50%. The anchor tag inside the content is styled to have no text decoration.

The last two blocks of code style the "view all" button. The button is given a width of 100px, a height of 35px, and a border radius of 5px. It has a background color of #1754a1 and white text with a font weight of 500. The border style is set to none. The "view all" section is a flex container and has the justify-content property set to center.

Overall, this CSS code is used to style the "trending-blogs" section of a web page. It sets the size and margin of the blog containers, the size of the blog images, and the style of the content and "view all" button.



The code creates a flex container with a column layout for the element with ID "products", and sets its child elements with class "row" to be flex containers with a row layout and evenly spaced items.

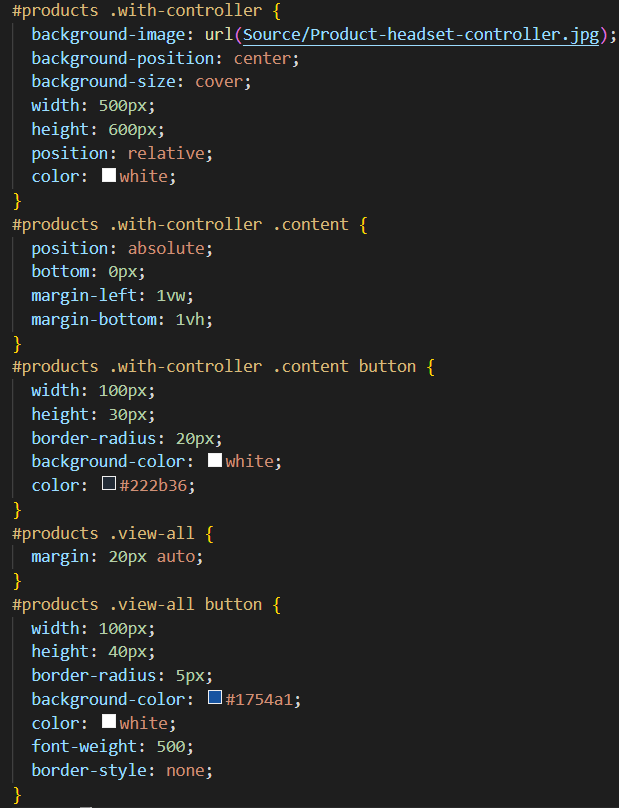


This code is related to styling for a product display section with an image and a button.

The first CSS rule sets a background image for the element with the class "without-controller", which is displayed as a product image. The image is centered and resized to cover the element's area. The width and height of this element are also set to specific values, and its position is set to relative.

The second CSS rule positions the content (in this case, a button) of the "without-controller" element absolutely at the bottom of the container.

The third CSS rule styles the button inside the "without-controller" element with specific width, height, and background color values. The button text is set to white.



This is a CSS code for a product section, where two different product cards are defined: one with a controller and another without a controller. The ".with-controller" and ".without-controller" classes define the background image of each card, as well as its dimensions and positioning.

The content of each card is positioned absolutely at the bottom left corner of the card, and a button is defined using the ".content button" class. The button has a white background color with black text for the "with-controller" card, and a dark blue background with white text for the "without-controller" card.

Lastly, a "view all" button is defined using the ".view-all button" class, which has a blue background color with white text, and is positioned at the center of the section using the ".view-all" class. The purpose of this button is to allow users to view all products in the section.



This code is for styling the footer section of a web page. The footer has a background color of #222b36 and white text. The details of the footer are arranged in a container with a class of "detail" and a width of 250px. The contact section of the footer has a class of "Contact" and is displayed using flexbox with space-evenly justification. The bottom section of the footer contains an h2 heading that is centered and has a padding of 10px. The social media links in the footer are styled using a class of "fonts" and are displayed using flexbox with space-evenly justification.