(1.) What are media queries in CSS and their use case?

Ans. Media queries are CSS techniques for applying different styles based on the media type and device characteristics such as screen size, screen resolution, and orientation.

=> The use case for media queries is to make responsive web design, where the layout adjusts to different devices and screen sizes.

The properties that can be used in media queries include:

1. min-width and max-width: match a media type based on the viewport width.
2. min-height and max-height: match a media type based on the viewport height.
3. min-device-width and max-device-width: match a media type based on the physical width of the device's screen.
4. min-device-height and max-device-height: match a media type based on the physical height of the device's screen.
5. orientation: match a media type based on the device orientation (portrait or landscape).
6. aspect-ratio: match a media type based on the aspect ratio of the viewport.
7. resolution: match a media type based on the device's screen resolution.
8. media type: matches a media type, such as 'print', 'screen', 'tv', etc.

=> To use media queries in CSS, you need to wrap your styles inside a media query block and specify the media type and conditions for which the styles should be applied.

Syntax:

```
@media <media-type> and (<condition>) {
    /* styles go here */
}
```

Example:

```
@media screen and (max-width: 600px) {
    /* styles for screens with a max width of 600px */
}
```

Note: You can specify multiple conditions separated by and. The styles inside the media query block will only be applied if all conditions are met.

(2.) Difference between min-width and max-width in media queries in CSS?

Ans. min-width and max-width are used in media queries to specify the minimum and maximum width of the viewport, respectively.

=> min-width specifies the minimum viewport width, below which the styles inside the media query block will not be applied.

=> max-width specifies the maximum viewport width, above which the styles inside the media query block will not be applied.

Example:

```
@media screen and (min-width: 600px) {
    /* styles for screens with a minimum width of 600px */
}
@media screen and (max-width: 600px) {
    /* styles for screens with a maximum width of 600px */
}
```

In this example, the styles inside the first media query block will only be applied to screens with a width greater than or equal to 600px, while the styles inside the second media query block will only be applied to screens with a width less than or equal to 600px.

(3.) Explain float property in CSS with examples?

Ans. The float property in CSS is used to position an element to the left or right of its parent container, allowing text and inline elements to wrap around it.

Syntax:

```
float: left | right | none | initial | inherit;
```

- left: The element is floated to the left of its parent container.
- right: The element is floated to the right of its parent container.
- none: The element is not floated (default value).
- initial: Sets the property to its default value.
- inherit: Inherits the float value from its parent element.

Example:

```
<div style="width: 100%;">
    <img src="image.jpg" style="float: right; width: 50px; height: 50px;">
    <p>This is some text that will wrap around the image on the right.</p>
</div>
```

In this example, the image is floated to the right and the text will wrap around it. Note that it is important to clear the float property for parent elements to avoid layout issues.

(4.) Explain clear property in CSS with examples?

Ans. The clear property in CSS is used to specify if an element should not be adjacent to floating elements on either the left or right side.

Syntax:

```
clear: left | right | both | none | initial | inherit;
```

- left: The element is not adjacent to floating elements on the left side.
- right: The element is not adjacent to floating elements on the right side.
- both: The element is not adjacent to floating elements on either the left or right side.
- none: The element is not restricted in terms of adjacent floating elements (default value).
- initial: Sets the property to its default value.
- inherit: Inherits the clear value from its parent element.

Example:

```
<div style="width: 100%;">
    <img src="image1.jpg" style="float: left; width: 50px; height: 50px;">
    <img src="image2.jpg" style="float: right; width: 50px; height: 50px;">
    <p style="clear: both;">This is some text that will not be affected by the floating images on either
        side.</p>
</div>
```

In this example, the first image is floated to the left and the second image is floated to the right. The text below is cleared with both, meaning it will not be adjacent to floating elements on either side.