Section 3
Introduction to CSS

Setting up the development environment

To start our class practice, first set the development environment in Android Studio. Create an ***Empty Activity*** with an activity **Name** as **Activity 3 – Type your last name**

Once Android Studio is set with an **assets folder**, we can open VS code text editor, create an **index.html** file and save it on **assets folder**.

Great! Now that we have our development environment set up, we can start working on our app layout using HTML.

Cascading Style Sheets, CSS

CSS (Cascading Style Sheets) allows you to create rules that specify how the content of an element should appear. For example, you can specify that the background of the page is cream, all paragraphs should appear in gray using the Arial typeface, or that all level one headings should be in a blue, italic, Times typeface.

CSS works by associating rules with HTML elements. These rules govern how the content of specified elements should be displayed. A CSS rule contains two parts: a **selector** and a **declaration.**

Selectors indicate which element the rule applies to. The same rule can apply to more than one element if you separate the element names with commas.

Declarations indicate how the elements referred to in the selector should be styled. Declarations are placed between brackets.

Declarations are split into two parts (a property and a value) are separated by a colon.

```
p {color: red;}
```

selector declaration

Within declaration

The declaration should specific the property and value of element of the selector.

Properties indicate the aspects of the element you want to change. For example, color, font, width, height and border. Each declaration ends with a semicolon. You can always refer to w3schools.com to check the list of all properties that is available in CSS.

Values specify the settings you want to use for the chosen properties. For example, if you want to specify a color property then the value is the color you want the text in these elements to be.

```
p {color: red;}
```

property color

Where should I place my CSS instructions?

Externally: Your CSS code can be writing in a text editor and save it with file extension .css

Then the .css can be found using the link element.

Internally: You can also include CSS rules within an HTML page by placing them inside a **<style>** element, which usually sits inside the <head> element of the page.

Using external CSS

The **<link>** element can be used in an HTML document to tell the browser where to find the CSS file used to style the page. It is an empty element (meaning it does not need a closing tag), and it lives inside the <head> element. It should use three attributes:

href: This specifies the path to the CSS file (which is often placed in a folder called css or styles).

type: This attribute specifies the type of document being linked to. The value should be text/css.

rel: This specifies the relationship between the HTML page and the file it is linked to. The value should be stylesheet when linking to a CSS file.

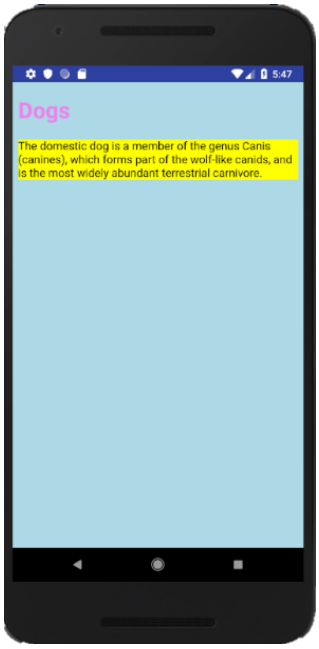
An HTML page can use more than one CSS style sheet. To do this it could have a <link> element for every CSS file it uses. For example, some authors use one CSS file to control the presentation (such as fonts and colors) and a second to control the layout.

Exercise) Prepare a cordova project folder (chapter 1). Open a text editor and save a HTML file as *index.*html in the www folder within the cordova project folder.

Now, in the HMTL file, create a simple app with **<h1>** and **<p>** element. Save an external .css file as ***style.css***

HTML

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Class Activity 3 by Huixin Wu</title>
    <link href="css/index.css" type="text/css" rel="stylesheet" />
  </head>
<body> <h1>Dogs</h1>
<p> The domestic dog is a member of the genus Canis (canines), which forms part of the
wolf-like canids, and is the most widely abundant terrestrial carnivore.</p>
</body>
</html>
```



index.css

```
body {background-color: lightblue;}
h1 {color:violet;}
p {background-color: yellow;}
```

Using Internal CSS

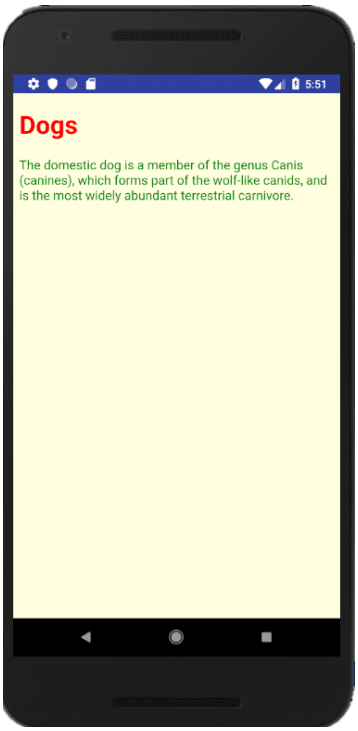
You can also include CSS rules within an HTML page by placing them inside a **<style>** element, which usually sits inside the <head> element of the page.

The **<style>** element should use the type attribute to indicate that the styles are specified in CSS. The value should be text/ css.

When building a site with more than one page, you should use an external CSS style sheet. This:

- Allows all pages to use the same style rules (rather than repeating them in each page).
- Keeps the content separate from how the page looks.
- Means you can change the styles used across all pages by altering just one file (rather than each individual page).

Exercise) Create the following text with font color.



HTML

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Class Activity 3 - CSS</title>
    <style type="text/css">
        body { background-color: lightyellow;}
        h1 { color: red;}
        p{ color: green;}
      </style>
  </head>
  <body> <h1>Potatoes</h1><p> There are dozens of different potato
  varieties. They are usually described as
  early, second early and main crop.</p>
  </body>
</html>
```

Text in CSS

The formatting of your text can have a significant effect on how readable your pages are.

Text Properties

The properties that allow you to control the appearance of text can be split into two groups:

- Those that directly affect the font and its appearance (including the typeface, whether it is regular, bold or italic, and the size of the text)
- Those that would have the same effect on text no matter what font you were using (including the color of text and the spacing between words and letters)

When choosing a typeface, it is important to understand that a browser will usually only display it if it's installed on that user's computer.

Typeface Terminology

SERIF

Serif fonts have extra details on the ends of the main strokes of the letters. These details are known as serifs.

In print, serif fonts were traditionally used for long passages of text because they were considered easier to read.

Examples: Georgia, Times New Roman

SANS-SERIF

Sans-serif fonts have straight ends to letters, and therefore have a much cleaner design.

Screens have a lower resolution than print. So, if the text is small, sans-serif fonts can be clearer to read.

Examples: Arial , Verdana, Helvetica

MONOSPACE

Every letter in a monospace (or fixed-width) font is the same width. (Non-monospace fonts have different widths.)

Monospace fonts are commonly used for code because they align nicely, making the text easier to follow.

Example: Courier

CURSIVE

Cursive fonts either have joining strokes or other cursive characteristics, such as handwriting styles.

Example: *Monotype Corsiva*

im

Exercise) Using the previous html code, create a new paragraph and name it as **author** using **class** property. Also, name the previous paragraph as **intro**

html file

Add new paragraph with class name **"author"**

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Class Activity 3 - CSS</title>
    <link href="css/index.css" type="text/css" rel="stylesheet" />
  </head>
  <body> <h1>Dogs</h1>
    <p class="author">By <i>Huixin Wu</i> </p>
    <p class="intro"> The domestic dog is a member of the genus Canis (canines),
    which forms part of the wolf-like canids, and is the most widely abundant
    terrestrial carnivore.</p>
</body>
</html>
```

index.css

```
body {
background-color: lightyellow;
}
h1 {
color:violet;
font-family: courier;
}
.intro{
color: green;
font-family: georgia;
}
.author{
color: blue;
font-family: "monotype corsiva";
}
```

Text Characteristics

font-weight

The font weight not only adds emphasis but can also affect the amount of white space and contrast on a page.

Examples:

Light Medium **Bold** Black

font-style

Italic fonts have a cursive aspect to some of the lettering. Oblique font styles take the normal style and put it on an angle

Examples:

Normal *Italic* *Oblique*

font-stretch

In condensed (or narrow) versions of the font, letters are thinner and closer together. In expanded versions they are thicker and further apart.

Examples:

Condensed Regular EXTENDED

Exercise) Using the previous code, add three more paragraphs with class name `fontWeight`, `fontStyle`, and `fontStretch`

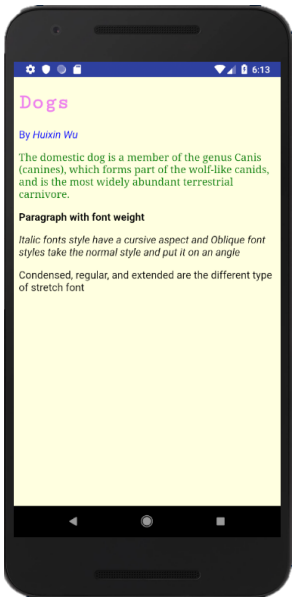
html file

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Class Activity 3 - CSS</title>
    <link href="css/index.css" type="text/css" rel="stylesheet" />
</head>
<body> <h1>Dogs</h1>
  <p class="author">By <i>Huixin Wu</i></p>
  <p class="intro"> The domestic dog is a member of the genus Canis (canines), which
forms part of the wolf-like canids, and is the most widely abundant terrestrial
carnivore.</p>
  <p class="fontWeight">Paragraph with font weight</p>
  <p class="fontStyle">Italic fonts style have a cursive aspect and Oblique font
styles take the normal style and put it on an angle</p>
  <p class="fontStretch">Condensed, regular, and extended are the different type of
stretch font</p>
</body>
</html>
```

Add the following paragraphs

style.css

```
.fontWeight{
  font-weight: bold;
}
.fontStyle{
  font-style: oblique;
}
.fontStretch{
  font-stretch: condensed;
}
```



Units of Text Type Size

The default size of text in a browser is 16 pixels.

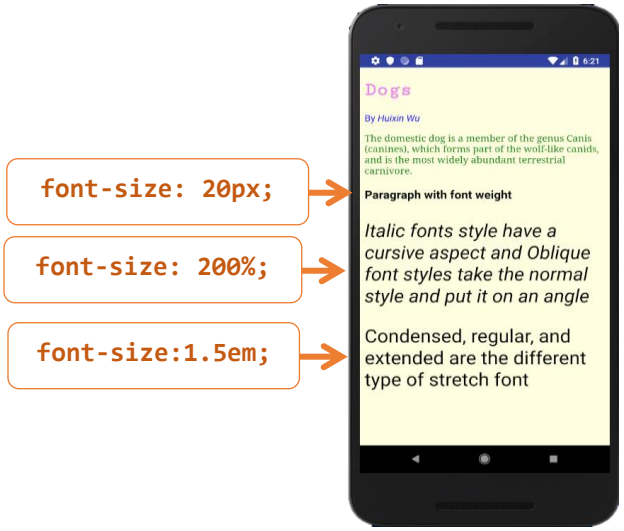
Text can be written as pixels, percentages, and ems.

Pixels	**Percentages**	**ems**
Setting font size in pixels is the best way to ensure that the type appears at the size you intended (because percentages and ems are more likely to vary if a user has changed the default size of text in their browser	The default size of text in a web browser is 16 pixels. Using percentages of this amount, you can create a scale where the default text size is 12 pixels, and headings are sized in relation to this.	Ems allow you to change the size of text relative to the size of the text in the parent element. Since the default size of text in web browsers is 16 pixels, you can use similar rules to those shown for percentages.
Pixels	**Percentages**	**Ems (Ephemeral Unit Scalable)**
body ➔16px h1 ➔ 32px h2 ➔ 24px h3 ➔18px	body ➔100% h1 ➔ 200% h2 ➔ 150% h3 ➔ 112.5%	body ➔100% p ➔1em h1 ➔2em h2 ➔1.5em h3 ➔1.125em

Exercise) Using the previous css file, use different values of `font-size` to class `fontWeight`, `fontStyle`, and `fontStretch`

style.css

```
.fontWeight{
  font-weight: bold;
  font-size: 20px;
}
.fontStyle{
  font-style: oblique;
  font-size: 200%;
}
.fontStretch{
  font-stretch: condensed;
  font-size: 1.5em;
}
```



Text Alignment (horizontal)

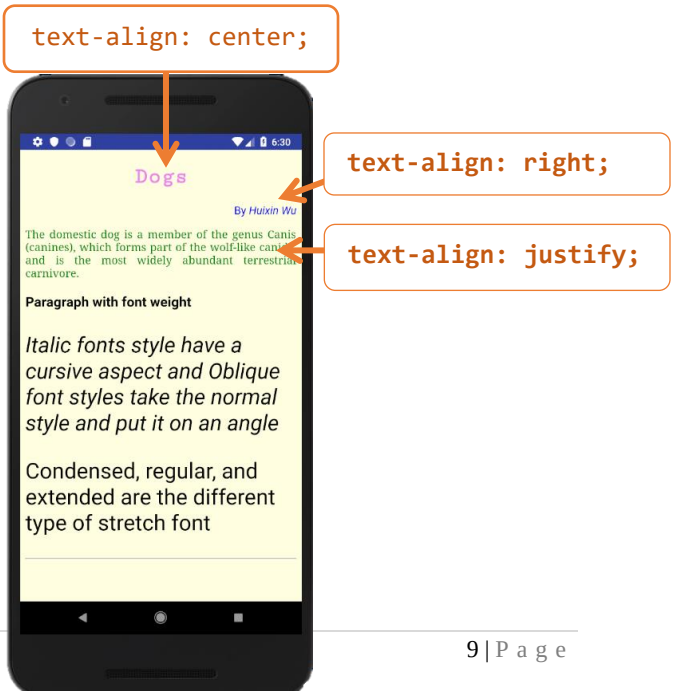
The **text-align** property allows you to control the alignment of text. The property can take one of four values:

- **left**: This indicates that the text should be left-aligned.
- **right**: This indicates that the text should be right-aligned.
- **center**: This allows you to center text.
- **justify**: This indicates that every line in a paragraph, except the last line, should be set to take up the full width of the containing box.

Activity) Using the previous css code, add different `text-align` properties to **`h1`**, **`.intro`**, and **`.author`**

index.css

```
h1 {
  color:violet;
  font-family: courier;
  text-align: center;
}
.intro{
  color: green;
  font-family: georgia;
  text-align: justify;
}
.author{
  font-family: "monotype corsiva";
  color: blue;
  text-align: right;
}
```



Text-Shadow

The **text-shadow** property is used to create a drop shadow, which is a dark version of the word just behind it and slightly offset. It can also be used to create an embossed effect by adding a shadow that is slightly lighter than the text.

text-shadow can use three lengths and a color for the drop shadow.

The first length indicates how far to the left or right the shadow should fall.

The second value indicates the distance to the top or bottom that the shadow should fall.

The third value is optional and specifies the amount of blur that should be applied to the drop shadow.

The fourth value is the color of the drop shadow.

```
text-shadow: 5px 10px 20px black;
```

Activity) Create different **text-shadow** effect.

Text Shadow

html

```
<h1 class="title1">Text Shadow</h1>
```

css

```
.title1{
  text-shadow: 10px 0px 0px black;
}
```

Text Shadow

html

```
<h1 class="title4">Text Shadow</h1>
```

css

```
.title4{
  text-shadow: 5px 10px 5px black;
}
```

Text Shadow

html

```
<h1 class="title2">Text Shadow</h1>
```

css

```
.title2{
  text-shadow: 0px 10px 0px black;
}
```

Text Shadow

html

```
<h1 class="title5">Text Shadow</h1>
```

css

```
.title5{
  text-shadow: 5px 10px 30px black;
}
```

Text Shadow

html

```
<h1 class="title3">Text Shadow</h1>
```

css

```
.title3{
  text-shadow: 5px 10px 0px black;
}
```

Text Border, Margin & Padding

Every box has three available properties that can be adjusted to control its appearance:

border

Every box has a border (even if it is not visible or is specified to be 0 pixels wide). The border separates the edge of one box from another.

Border Style

You can control the style of a border using the border-style property. This property can take the following values:

- **solid** a single solid line.
- **dotted** a series of square dots (if your border is 2px wide, then the dots are 2px squared with a 2px gap between each dot)
- **dashed** a series of short lines
- double two solid lines (the value of the border-width property creates the sum of the two lines)
- **groove** appears to be carved into the page
- **ridge** appears to stick out from the page
- **inset** appears embedded into the page
- **outset** looks like it is coming out of the screen
- **hidden / none** no border is shown

border values can be written in one line: border style, border size, and border color:

```
border: dashed 5px red;
```

Activity) Add border style, size, and color to a paragraph

html

```
<p class="page_border">How to work with border</p>
```

css

```
.page_border{
  border: dashed 5px red;
  font-size: 30px;
}
```

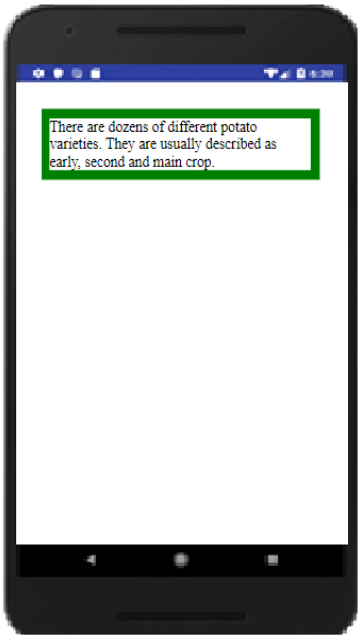
output

margin

Margins sit outside the edge of the border. You can set the width of a margin to create a gap between the borders of two adjacent boxes.

The margin property controls the gap between boxes. Its value is commonly given in pixels, although you may also use percentages or ems.

Example) Add the same margin to add side of a paragraph



html

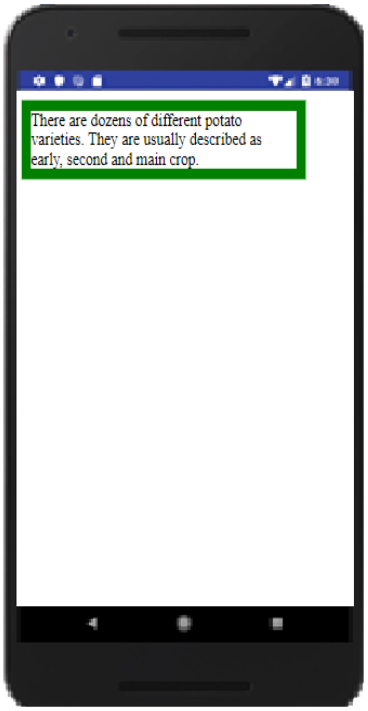
```
<p class="m1">There are dozens of
different potato varieties. They are
usually described as early, second and
main crop.</p>
```

css

```
.m1{margin: 30px;
    border: solid 10px green;}
```

You can specify values for each side of a box using: margin-top, margin-right, margin-bottom, margin-left.

Example) Add right margin to a paragraph



html

```
<p class="m2">There are dozens of
different potato varieties. They are
usually described as early, second and
main crop.</p>
```

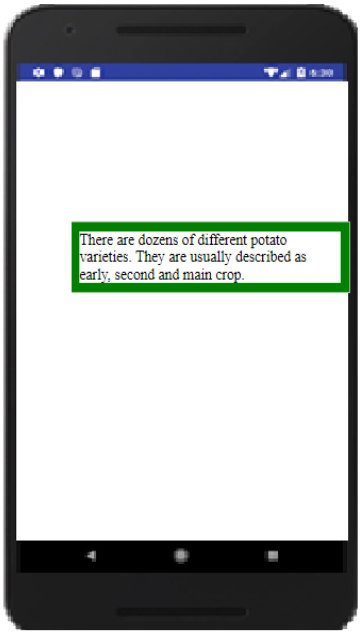
css

```
.m2{margin-right:50px;
    border: solid 10px green;}
```

You can also use the shorthand (where the values are in clockwise order: top, right, bottom, left):

```
margin: 100px 0px 50px 30px;
```

Example) add the following four margins to a paragraph: margin top: 100px, margin bottom: 50px, margin left 30px, and 0px of margin right



html

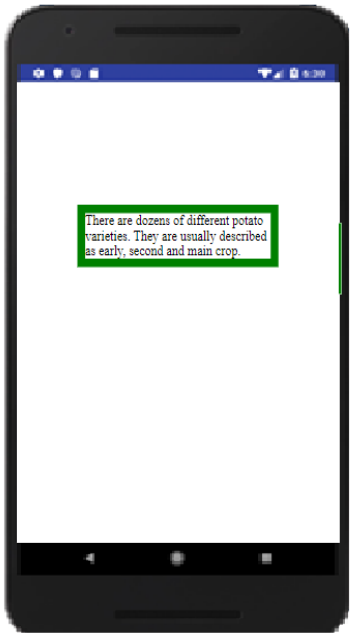
```
<p class="m3">There are dozens of
different potato varieties. They are
usually described as early, second and
main crop.</p>
```

css

```
.m3{margin: 100px 0px 50px 30px;
    border: solid 10px green;}
```

Another shorthand to add margin is by given two values. The first value is for the top and bottom margin, and the second value is for the left and right margin.

Example) add the following margin to a paragraph: 100px to top and bottom margin, and 50px to left and right margin:



html

```
<p class="m4">There are dozens of
different potato varieties. They are
usually described as early, second and
main crop.</p>
```

css

```
.m4{margin: 100px 50px;
    border: solid 10px green;}
```

padding

Padding is the space between the border of a box and any content contained within it. Adding padding can increase the readability of its contents.

The padding property allows you to specify how much space should appear between the content of an element and its border.

The value of this property is most often specified in pixels (although it is also possible to use percentages or ems). If a percentage is used, the padding is a percentage of the browser window (or of the containing box if it is inside another box).

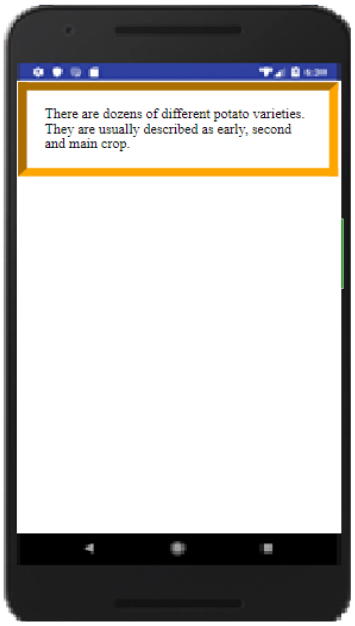
Example) add 20px of padding to all the side of a paragraph

html

```
<p class="padding1">There are dozens of
different potato varieties. They are
usually described as early, second and
main crop.</p>
```

css

```
.padding1{padding: 20px;
border: 10px inset orange; }
```



You can specify different values for each side of a box using: padding-top; padding-right; padding-bottom; padding-left.

Example) add 30px of left padding to a paragraph

html

```
<p class="padding2">There are dozens of
different potato varieties. They are
usually described as early, second and
main crop.</p>
```

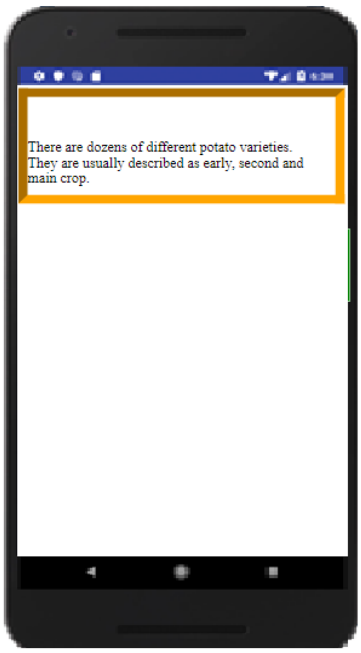
css

```
.padding2{padding-left: 30px;
border: 10px inset orange; }
```

Or you can use a shorthand (where the values are in clockwise order: top, right, bottom, left):

```
padding: 50px 20px 10px 0px;
```

Example) add the following four paddings to a paragraph: padding top: 50px, padding bottom: 10px, padding right 20px, and 0px of left padding.



html

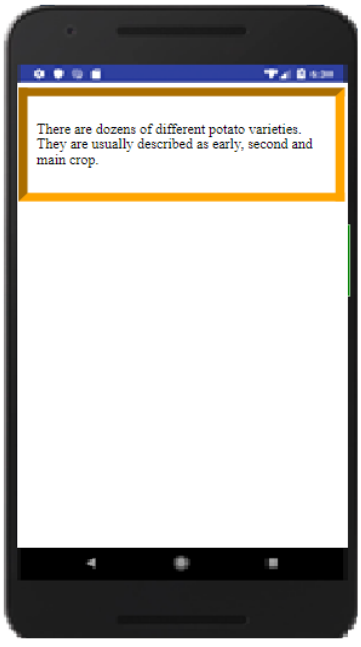
```
<p class="padding3">There are dozens of
different potato varieties. They are
usually described as early, second and
main crop.</p>
```

css

```
.padding3{padding: 50px 20px 10px 0px;
border: 10px inset orange; }
```

Another shorthand to add padding is by given two values. The first value is for the top and bottom padding, and the second value is for the left and right padding.

Example) add the following padding to a paragraph: 50px to top and bottom padding, and 30px to left and right padding:



html

```
<p class="padding4">There are dozens of
different potato varieties. They are
usually described as early, second and
main crop.</p>
```

css

```
.padding4{margin: 50px 30px;
    border: solid 10px green;}
```