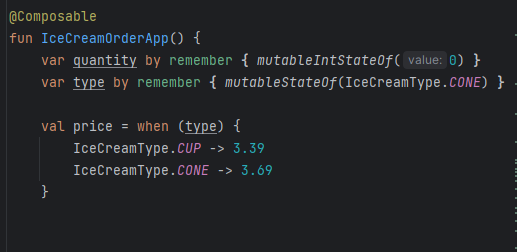
Danny Trenholm

201202966

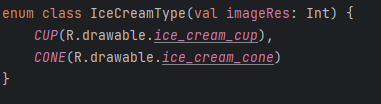
[X2012cml@stfx.ca](mailto:X2012cml@stfx.ca)

Feb 13th, 2024

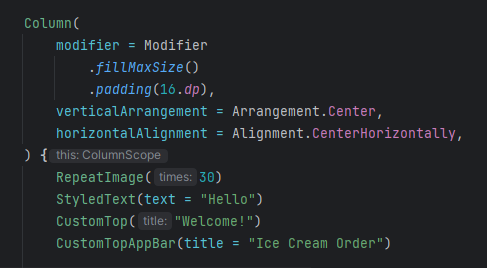
Report- Assignment#1



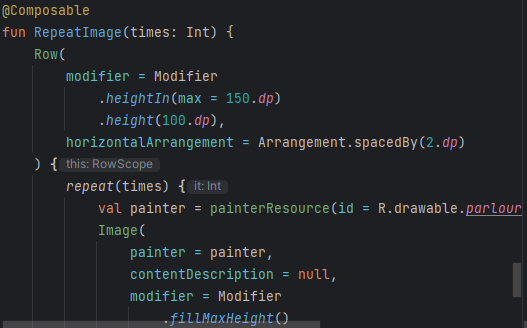
The above is using two defining types which are CUP and CONE from this enum class.

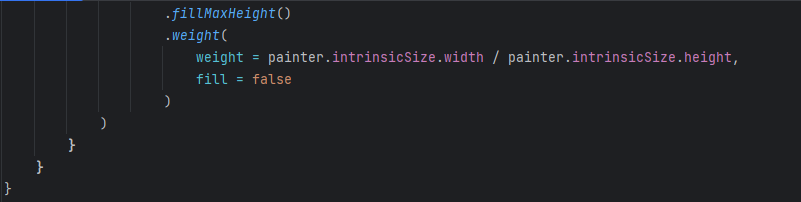


The enum iterates from the two values, anyways, continuing…



The above continues on IceCreamOrderApp(), in which you may notice 4 custom jetpack compose elements, RepeatImage(30), StyledText(text = “Hello”), CustomTop(“Welcome!”), and CustomTopAppBar(title = “Ice Cream Order”), we will explore them in sequence.

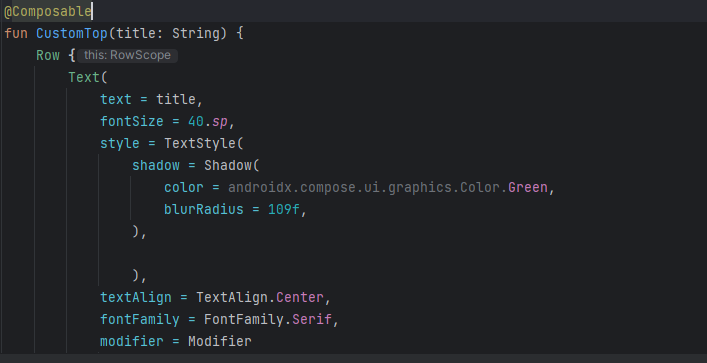


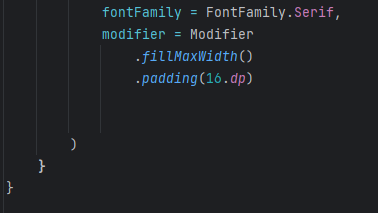


RepeatImage takes a Row, and makes a repeating function that paints a border at the top of the page. The image is based off of a barber shop spinning post. It is repeated 30 times



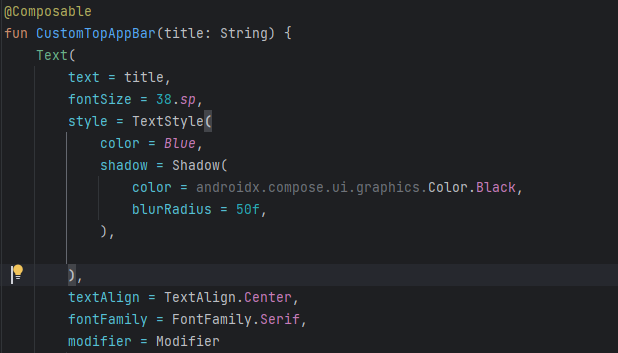
StyledText makes custom letter colors, font sizes, and odd bold letters for the word “Hello”. H is Red, E is Green, l is Blue, and o is yellow The text is then put to string after the characters are generated in the row.

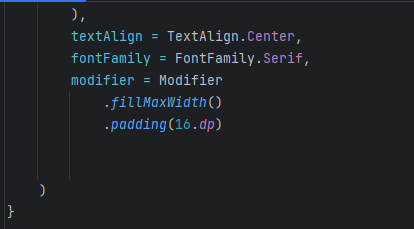




CustomTop() is used for text manipulation purposes, and changes things such as the font family, font size, style (for shadows, color, and blur effect around the letters. It is centered in the middle.

Similar to CustomTop(), there is CustomTopAppBar()

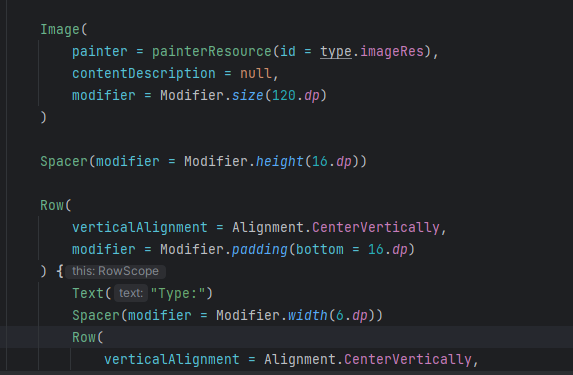




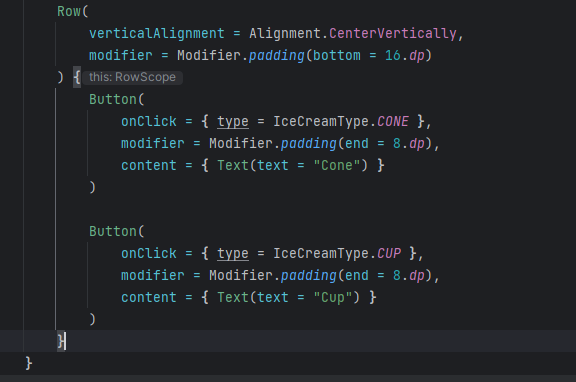
There is slight differences, with color and size, but that is pretty much it.



Now, to continue to the main screen function.



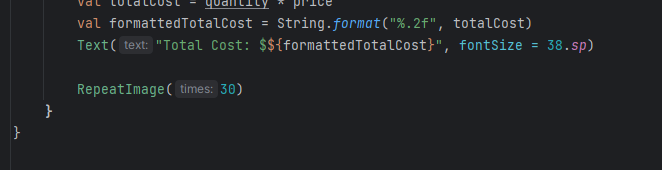
The Image is important to note that the type.imageRes is basically switching between the enum classes, and simply changes the picture accordingly. There are some simple styles made with spacers in the colum based architecture, and Type likes up with these buttons



The buttons change the type based on a click, from CONE, and CUP.

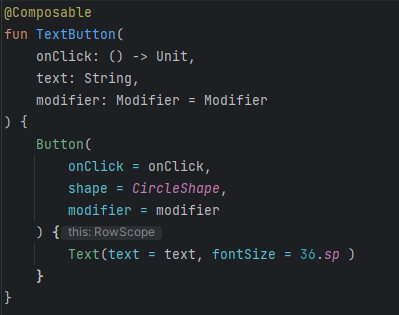




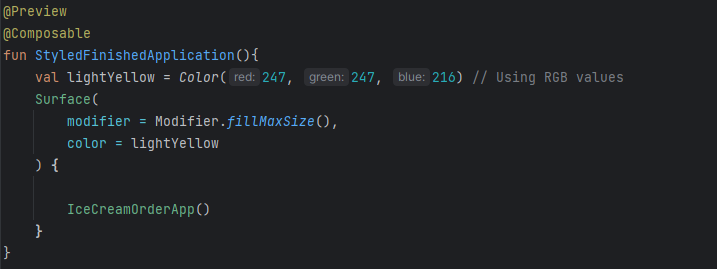


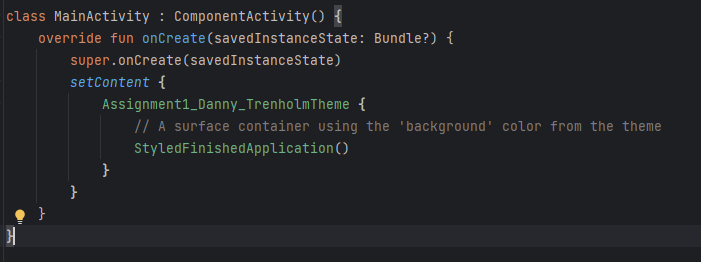
There is a new jetpack composable here, called TextButton. It is just so the – and + button line up with the quantity increasing and decreasing in the Text, where quantity.toString() puts the total into it.. Below is the calculacation at totalCost, that is turned to formattedTotalCost to put a 2nd digit past the decimal. Then the RepeatImage() is called at the end again for the bottom border.

Here is the TextButton component, which simply just contains a Text wrapped into a Button



Then the final wrap up. A custom background in the surface, which is an RGB value of light yellow just to give it a nicer look.





The import statements were all added with ALT+Enter.

