rCharts

- rCharts is a way to create interactive javascript visualization using R
- So

1. You don’t have to learn complex tools, like D3

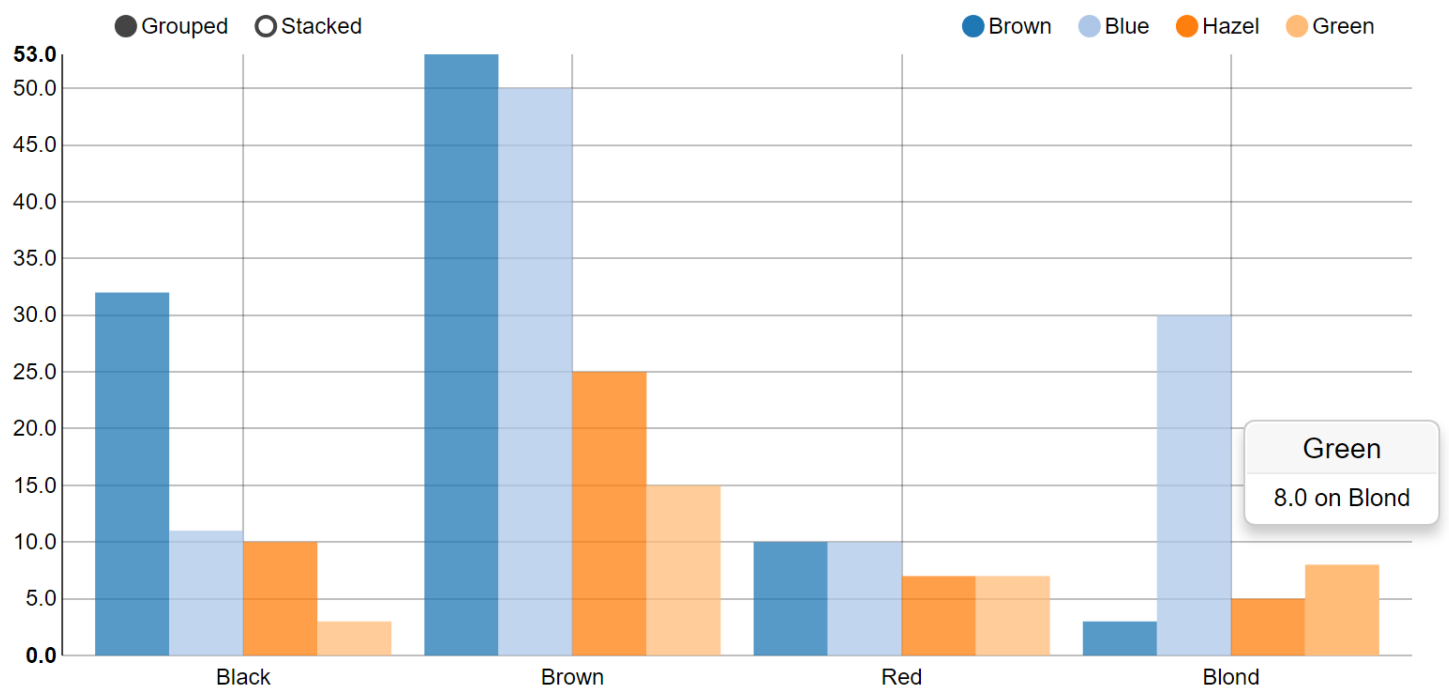
D3.js is a JavaScript library for manipulating documents based on data. **D3** helps you bring data to life using HTML, SVG, and CSS. D3’s emphasis on web standards gives you the full capabilities of modern browsers without tying yourself to a proprietary framework, combining powerful visualization components and a data-driven approach to DOM manipulation**.**

2. You simply work in R learning a minimal amount of new syntax

- rCharts was written by Ramnath Vaidyanathan, who also wrote slidify
- rCharts uses a formula interface to specify plots, just like the lattice package (https://www.statmethods.net/advgraphs/trellis.html).

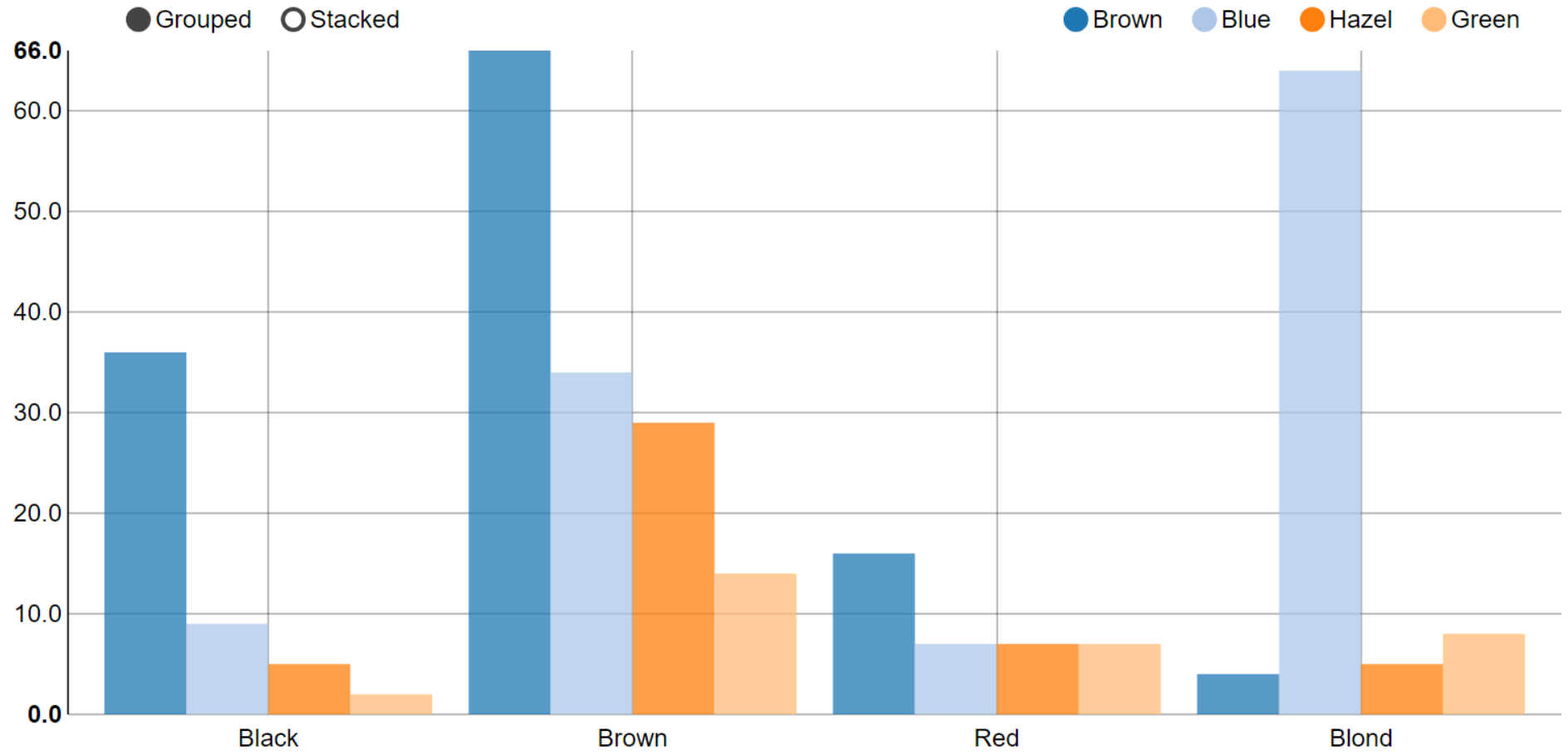
Example

```
hair_eye = as.data.frame(HairEyeColor)
n1<- nPlot(Freq ~ Hair , group= 'Eye', type = 'multiBarChart', data=subset(hair_eye,
Sex=='Male'))
n1
```



Example

```
hair_eye = as.data.frame(HairEyeColor)
n1<- nPlot(Freq ~ Hair , group= 'Eye', type = 'multiBarChart', data=subset(hair_eye,
Sex==‘Female'))
n1
```



Viewing the plot

The object n1 contains the plot
In Rstudio, typing n1 brings up the plot in the Rstudio viewer (or you can just not assign it to an object)
Do n1 $ then hit TAB to see the various functions contained in the object

- n1$html() prints out the html for the plot

I do n1$save(filename) then bring the code back into slidify document
This is recommended for slidify, but if you are just looking at the plot, its unnecessary