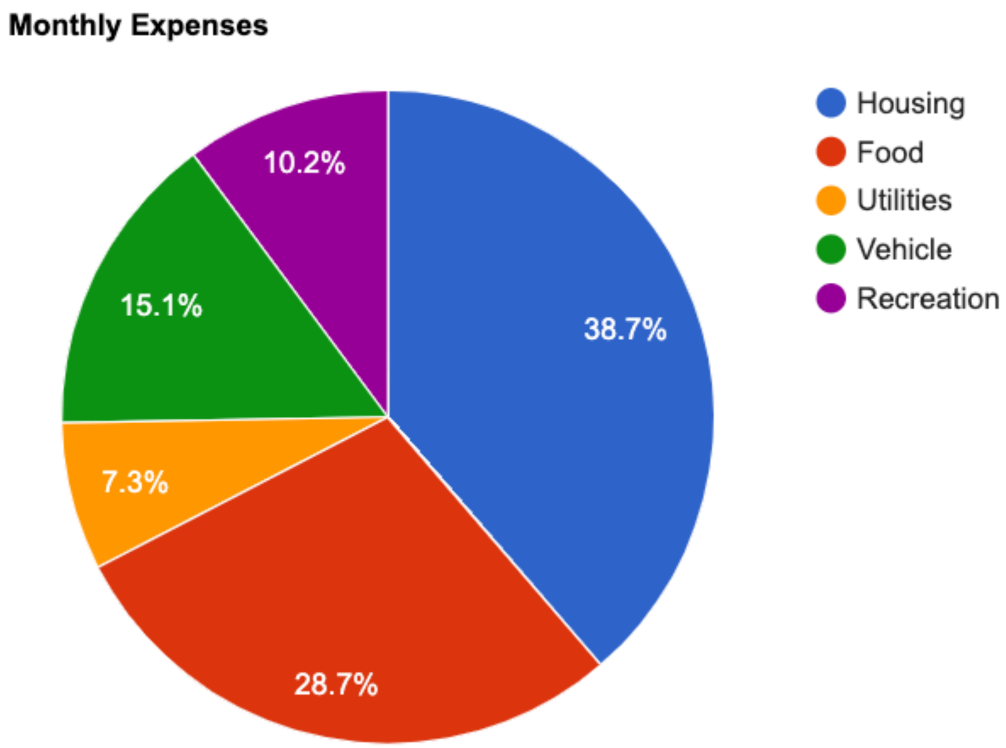
WISE GUIDES
CAREER CHANGER

Rev. 25Jan2019

Exercise 35

TIME FOR A TREAT AND A PRIZE!

After those last two exercises, you deserve some fun. And nothing says "fun" more than some cool-looking charts. Let's start with a piechart:



When you want to create charts, you're going to look for a third-party library to use. There are lots of them, and some are very expensive. But *this* piechart comes to us for free thanks to Google.

To make use of the chart, add this `<script>` tag right above the one already in `index.html`:

```
<script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
```

That gets us the base "engine", so to speak. Now to customize the chart in our `index.js`.

Good third-party libraries strive to make using them as easy as possible. The tradeoff for that is that you'll write some code whose purpose you have no idea of.

□ That's true of the first line of code you'll write:

```
google.charts.load('current', {'packages': ['corechart']})
```

□ The next snippet you'll write will be a little more understandable:

The values I've used are arbitrary. You might want to use the ones for your household

```
let drawChart = () => {
  let data = google.visualization.arrayToDataTable(
    [
      ['Expense', 'Amount Spent'],
      ['Housing', 1575],
      ['Food', 1168],
      ['Utilities', 298],
      ['Vehicle', 615],
      ['Recreation', 414],
    ]
  )
}
```

□ Next, provide the title of the piechart:

```
let options = {
  title: 'Monthly Expenses'
}
```

□ And some more code that we don't understand much of:

```
let chart = new google.visualization.PieChart(document.getElementById('piechart'))
chart.draw(data, options)
google.charts.setOnLoadCallback(drawChart)
```

□ And with that, you're ready to run your repl.

That's a *lot* of code that we really don't understand. Is that OK? Yes. Obviously, you need to trust the source of the third-party code you use, but very, very few developers are going to dig into their code to understand exactly how it works. (That's sort of the point of using third-party libaries — so that we can just use it without going through the effort of writing it ourselves.)

Good third-party libraries do provide you with documentation so that you can further customize the output.

Often, this is done in the context of an options object and, in fact, you wrote just such an object:

```
let options = {
  title: 'Monthly Expenses'
}
```

□ Google offers several more options that you can include in the `options` object. One cool one turns the 2D piechart into a 3D piechart by adding this key:value pair to the `options` object: `is3D: true`. Try it out!

□ Another option produces a donut chart. To see this, remove the `is3D` key:value pair and insert this one instead: `pieHole: 0.4`.

Interested in exploring more? Checkout the Google documentation at: https://developers.google.com/chart/interactive/docs/gallery/piechart