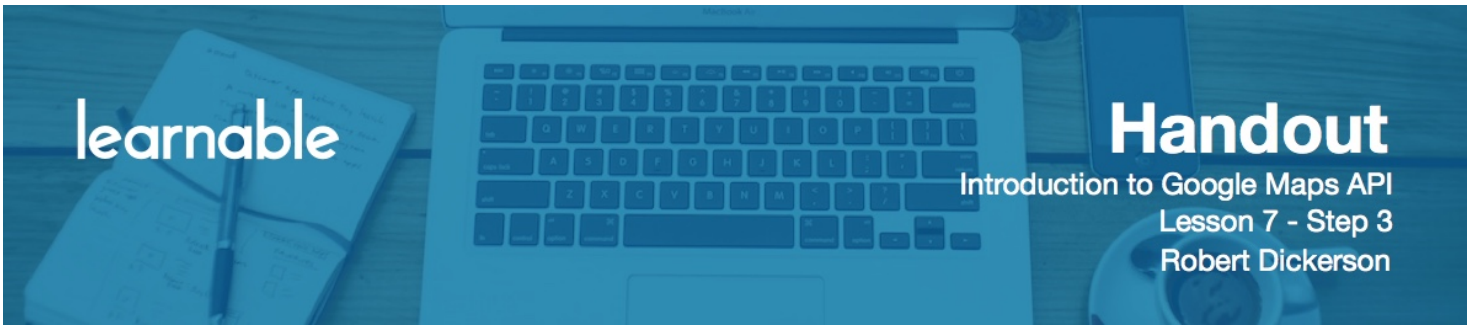

Introduction

At this final step we will finish the course project, and turn it into a compelling visualization.

We will use four different colored airport icons to identify the airports from best to worst based on airport on-time performance. I've chosen to represent the green airplane as airports that have an average arrival and departure greater than 80%. The yellow icon represents on-time performance between 70 and 79, orange between 60 and 69, and red is less than 60%. I chose these break points arbitrarily to get a good color distribution, although frequent travellers might disagree that 80% is a good performing airport.

Change Icons colors to reflect performance

Let's change the color of the icons to reflect the on-time performance:

```
if(airport.totalper >= 80) {
  airport.icon = 'green';
}
else if((70 <= airport.totalper) && (airport.totalper < 80)) {
  airport.icon = 'yellow';
}
else if((60 <= airport.totalper) && (airport.totalper < 70)) {
  airport.icon = 'orange';
}
else {
  airport.icon = 'red';
}
```

We set the icon based on these conditions.

Resize icons based on number of operations

Now let's adjust icon's size based on number of operations. I've chose some arbitrary break points here.

```
if(airport.totalflights > 10000) {
  airport.iconsize = new google.maps.Size(48,48);
}
else if((1000 <= airport.totalflights) && (airport.totalflights <= 10000)) {
  airport.iconsize = new google.maps.Size(32,32);
}
else if(airport.totalflights < 1000) {
  airport.iconsize = new google.maps.Size(16,16);
}
```

Once again we use a series of if-else statements that will determine based on the total flights, the icon size that we want.

Lastly update the `icon` property inside the `addMarker`:

```
icon: {
  url: 'img/airplane-'+airport.icon+'.png',

  size: airport.iconsize,

  origin: new google.maps.Point(0,0),

  anchor: new google.maps.Point(16,32),
```

```
  scaledSize: airport.iconsize
}
```