

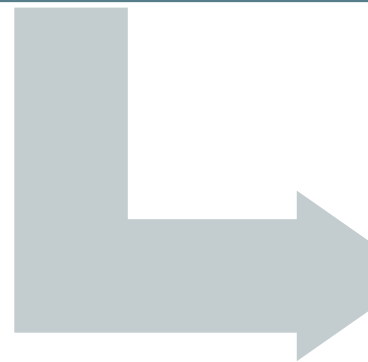
Statistics

Steps to follow:

1. Add code to perform calculation in controller file



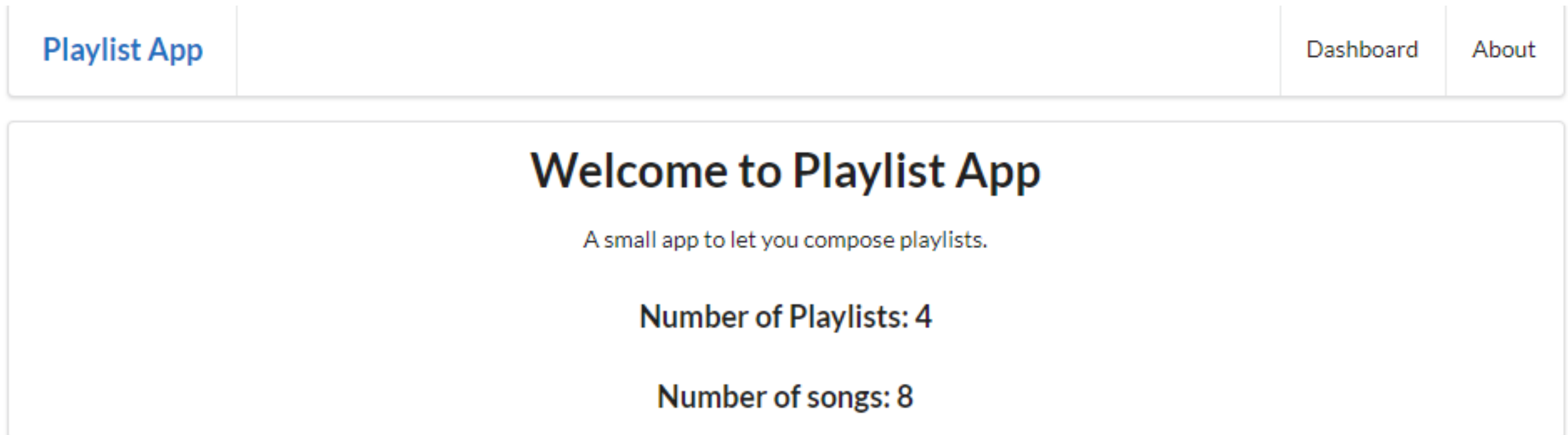
2. Pass results to view (by placing in viewData object)



3. Display results in view, using handlebars expressions

Example:

- In this example, we will add statistics to the **start** view
 - Total number of playlists
 - Total number of songs



1. Add code to perform calculation in controller file

- In controllers/start.js:
 - Use getAllPlaylists() method to get playlists
 - Loop through collection, counting songs in each playlist as shown here

```
6 ✓ const start = {
7 ✓   index(request, response) {
8     logger.info('start rendering');
9
10    const playCollections = playlistStore.getAllPlaylists();
11    let totalsongs = 0;
12 ✓   for (let i in playCollections) {
13     totalsongs = totalsongs + playCollections[i].songs.length;
14   }
15 }
```

2. Pass results to view (by placing in viewData object)

- The viewData object contains information to be passed to the view. Add two properties to this object:
 - totalsongs, which stores the totalsongs variable just created in previous step
 - totalCollections, containing the length of our playCollections array (i.e. the number of playlists)

```
const viewData = {  
  title: 'Welcome to Playlist App',  
  totalCollections: playCollections.length,  
  totalsongs: totalsongs,  
};  
response.render('start', viewData);  
},
```

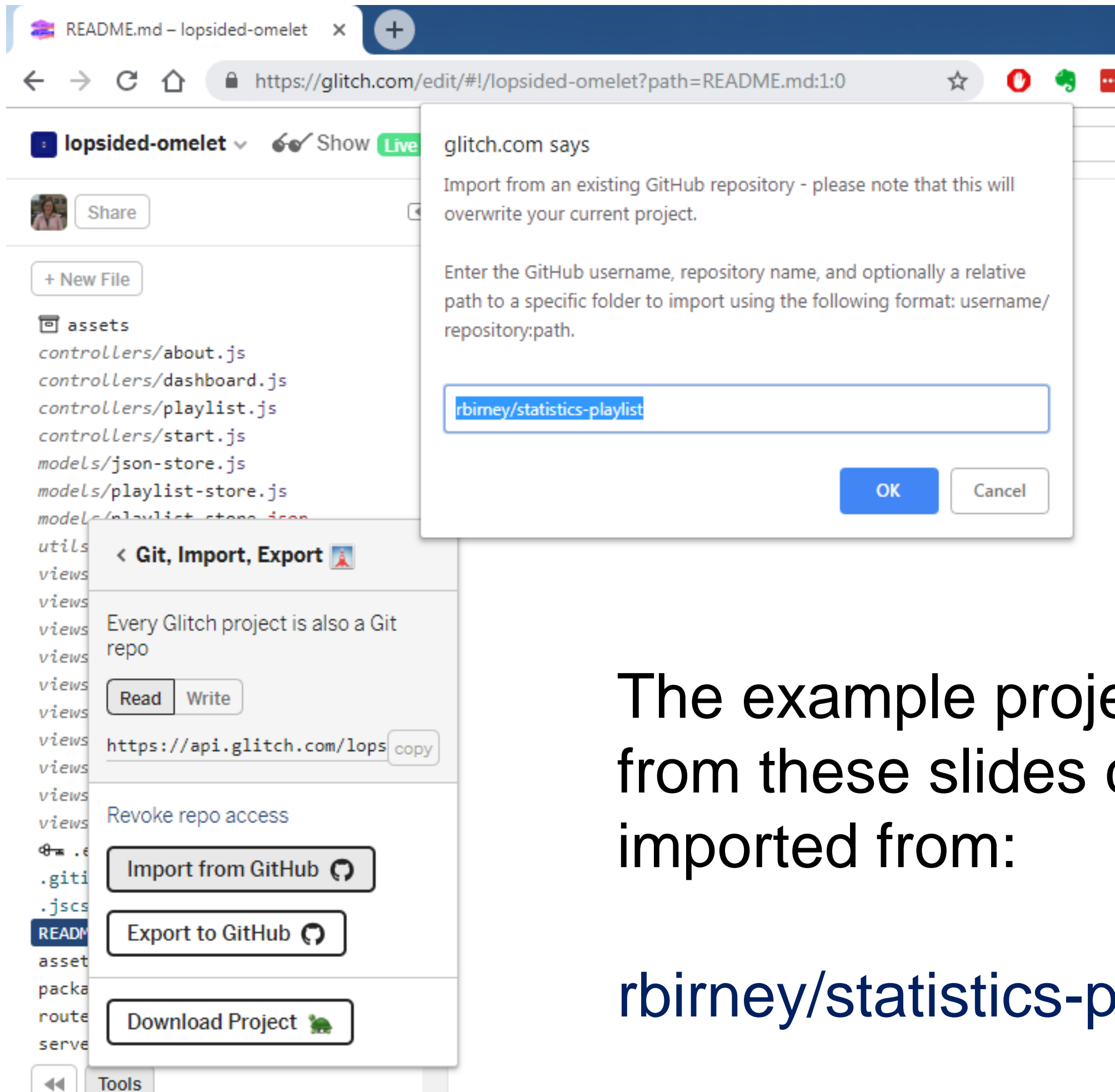
```
1  'use strict';
2
3  const logger = require('../utils/logger');
4  const playlistStore = require('../models/playlist-store');
5
6  const start = {
7    index(request, response) {
8      logger.info('start rendering');
9
10     const playCollections = playlistStore.getAllPlaylists();
11     let totalsongs = 0;
12     for (let i in playCollections) {
13       totalsongs = totalsongs + playCollections[i].songs.length;
14     }
15
16     const viewData = {
17       title: 'Welcome to Playlist App',
18       totalCollections: playCollections.length,
19       totalsongs: totalsongs,
20     };
21     response.render('start', viewData);
22   },
23 };
24
25 module.exports = start;
26
```

Complete
controllers/start.js
now looks like this

3. Display results in view, using handlebars expressions

- views/start.hbs
 - Display totalCollections and totalsongs as handlebars expressions (in double curly brackets)

```
1  {{> menu}}
2
3  <section class="ui center aligned middle aligned segment">
4    <h1 class="ui header">
5      Welcome to Playlist App
6    </h1>
7    <p>
8      A small app to let you compose playlists.
9    </p>
10
11   <h3>Number of Playlists:  {{totalCollections}} </h3>
12   <h3>Number of songs:    {{totalsongs}}</h3>
13
14  </section>
```



The example project from these slides can be imported from:

rbirney/statistics-playlist