5.4 Data Visualisation

5.4.1 Introduction / Recap

Data Visualisation

5.4.1 Introduction / Recap

- During the last two sessions we have been looking at collaboration

- This session we will be looking at how to use data from external sources to create interactive visualisations

5.4.1 Introduction / Recap

- You will learn
  - How to set up user interfaces that display fields from the database
  - How to query data inserted in the database and display it on the page
  - How to render data from database queries and visualise them in interesting ways

5.4.2 Setting up the Interface

Data Visualisation

5.4.2 Setting up the Interface

- Demo of the application

5.4.2 Setting up the Interface

The main container :

```
<div class="container">

  {{> song_viz}}

</div>
```

5.4.2 Setting up the Interface

```
<div class=“row"> // this is a two column setup

   <div class=“col-md-6"> // First column

      {{> song_viz_controls}}

        {{> song_feature_list}} // Feature list

   </div>
```

5.4.2 Setting up the Interface

```
<div class=“col-md-6"> // Second column

     {{> song_vizjs}} // the visualisation

</div>
```

5.4.2 Setting up the Interface

```
// Then in .js

Template.song_viz_controls.events({

    "change .js-select-single-feature”:function(event){

      event.preventDefault();

      var feature = $(event.target).val();

      Session.set("feature", {name:feature, type:"single_features"});

    },
```

5.4.2 Setting up the Interface

- Helper functions for the feature list display template
- Helper functions for the vis control form
- Event handlers for the viz control form

5.4.3 Getting Data

Data Visualisation

5.4.3 Getting Data

```
<select class="js-select-single-feature">
  {{#each get_feature_names "single"}}
  <option value="{{name}}">{{name}}</option>
  {{/each}}
 </select>
```

5.4.3 Getting Data

```
Template.song_viz_controls.events({

"change .js-select-single-feature”:function(event){

event.preventDefault();

var feature = $(event.target).val();

Session.set("feature", {name:feature,
type:”single_features"});

}
```

5.4.3 Getting Data

- We are using a reactive session
  - Once the feature selection is made, all other event handlers refer to the session for data
  - This includes the text
  - It also includes the data visualisation

5.4.4 Rendering Data on the page

Data Visualisation

5.4.4 Rendering Data on the page

- There are two functions for rendering the visualisation

```
function initDateVis(){

// This is the timeline

function initBlobVis(){

// This is the blobs
```

5.4.4 Rendering Data on the page

- Clear everything
- Generate an array of items from the songs collection
- Iterate the songs collection, converting each song into a simple object that the visualiser understands
- Construct a datapoint for each song and stick it in an array
- Pass this to the vis.js library

5.4.4 Rendering Data on the page

- Different visualisers require slightly different initialisation
- Take a look at how the two functions are different to understand how to set up new ones.

5.4.5 Review

Data Visualisation

5.4.5 Review

- In this session we learned
  - How to set up user interfaces that display fields from the database
  - How to query data inserted in the database and display it on the page
  - How to render data from database queries and visualise them in interesting ways

5.4.5 Things to try

- Different types of visualisation
- Plot types of data against each other
- Plot results based on conditionals

Bye!