**CODECLAN**

**WK7 DAY3 HW**

**CONTINENTS APP**

Apologies for narrative instead of drawings – hope this suffices………

* **Looking at the codebase, draw a diagram to visualise the Pub/Sub event and data flow through the app.**

*API data enters app at continents.js as an array of country objects (see example below).*

*Continents.js publishes (continents:continents-data-ready) extracts from that data as produced by functions getContinentNames (list of continents derived by looping through countries and compiling unique continent name list) and modelContinents (continent name and all countries of that continent).*

*Continent\_list\_view.js subscribes to continents:continents-data-ready.*

*Example:*



* **Looking at the codebase, draw a tree-diagram to show the view rendering responsibility hierarchy.**

*Continent\_list\_view.js renders a new ContinentView for each continent*

*Continent\_view renders continent name (h2) and each country for that continent (li)*

* **Using the browser Inspect pane in the devtools to view the DOM, draw out the structure of the web page. Use the IDs and classes as the labels.**



*The continent class is repeated for each continent.*

* **Answer the following questions:**

1. **Which view is rendered more than once?**

*Continent\_view.js*

1. **Which views are nested in which other view?**

*Continent\_view.js is nested within continents\_list\_view.js*

1. **What are the benefits of using nested reusable views in this way?**

*Benefit of DRY code (speed, ease of maintenance, reduced storage, etc)*