

Week08 - Debugging, Formatter, Fragment

Name - Priyabrata Priyadarshi

1.

```
async function getUserAsync(name) {
  try {
    let response = await fetch(`https://api.github.com/users/${name}`);
    return await response.json();
  } catch(err) {
    console.error(err);
  }
}
```

2.

The screenshot shows a web browser's developer tools interface. The 'Sources' tab is active, displaying the file 'App.controller.js'. The code is as follows:

```
1 sap.ui.define([
2   "sap/ui/core/mvc/Controller",
3   "sap/ui/core/library",
4   "sap/m/Dialog",
5   "sap/m/Button",
6   "sap/m/library",
7   "sap/m/Text"
8 ],
9
10 /**
11  * @param {typeof sap.ui.core.mvc.Controller} Controller
12  */
13 function (Controller, coreLibrary, Dialog, Button, mobileLibrary, Text) {
14   "use strict";
15
16   // shortcut for sap.m.ButtonType
17   var ButtonType = mobileLibrary.ButtonType;
18
19   // shortcut for sap.m.DialogType
20   var DialogType = mobileLibrary.DialogType;
21
22   // shortcut for sap.ui.core.ValueState
23   var ValueState = coreLibrary.ValueState;
24
25   return Controller.extend("io.pb.q6button.controller.App", {
26     onInit: function () {
27
28     },
29
30     onSuccessMessageDialogPress: function () {
31       debugger;
32       if (!this.oSuccessMessageDialog) {
33         this.oSuccessMessageDialog = new Dialog({
34           type: DialogType.Message,
35           title: "Success",
36           state: ValueState.Success,
37           content: new Text({ text: "Success" }),
38           beginButton: new Button({
39             type: ButtonType.Emphasized,
40             text: "OK",
41             press: function () {
42               this.oSuccessMessageDialog.close();
43             }.bind(this)
44           })
45         });
46
47         this.oSuccessMessageDialog.open();
48       }
49     });
50   });
51 }
```

The right-hand pane shows the 'Debugger paused' state. The 'Scope' section lists the following variables:

- Local: this: f
- Closure
- Global: Window

The 'Call Stack' section shows the following stack:

- onSuccessMessageDialogPress (App.controller.js:30)
- b.fireEvent (EventProvider-dbg.js:247)
- d.fireEvent (Element-dbg.js:626)
- (anonymous) (ManagedObjectMe...ata-dbg.js:789)
- k.onTap (Button-dbg.js:542)
- d._handleEvent (Element-dbg.js:342)
- U._handleEvent (UIArea-dbg.js:947)
- dispatch (jquery-dbg.js:5430)
- g (jquery-mobile-c...om-dbg.js:1899)
- q (jquery-mobile-c...om-dbg.js:2019)
- dispatch (jquery-dbg.js:5430)
- t3.handle (jquery-dbg.js:5234)
- trigger (jquery-dbg.js:8823)
- (anonymous) (jquery-dbg.js:8901)
- each (jquery-dbg.js:385)
- each (jquery-dbg.js:207)
- trigger (jquery-dbg.js:8900)

3.

```
> function reverseString(str) {
  let newString = "";

  for (var i = str.length - 1; i >= 0; i--)
  {
    newString += str[i];
  }
  return newString;
}

console.log(reverseString('this string is
reversed'));

desrever si gnirts siht VM87:11
```

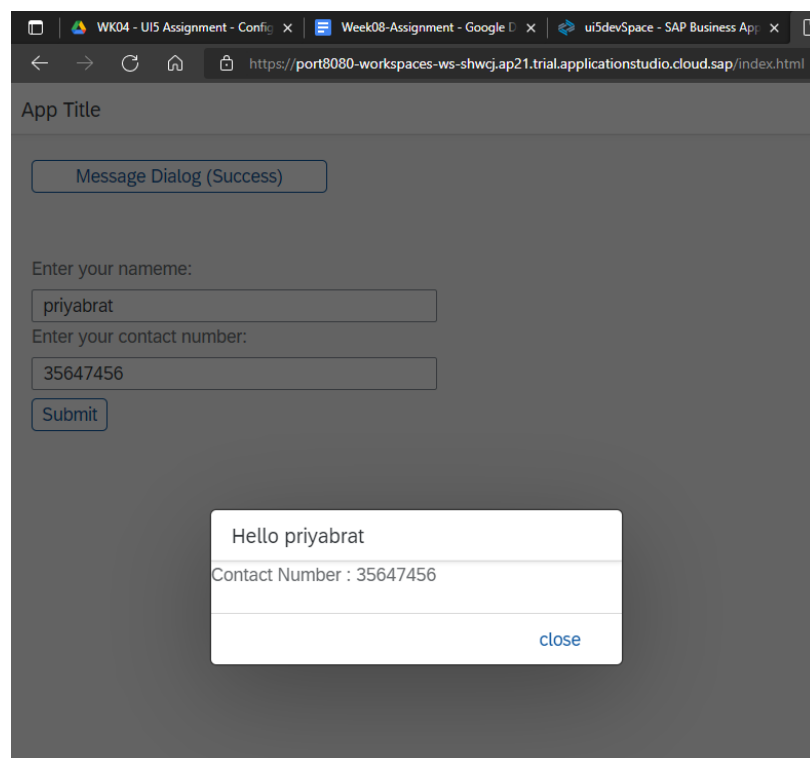
```
> function bubbleSort(arr) {
  for (var i = 0; i < arr.length; i++) {
    for (var j = 0; j < (arr.length - i -
1); j++) {
      if (arr[j] > arr[j + 1]) {
        var temp = arr[j]
        arr[j] = arr[j + 1]
        arr[j + 1] = temp
      }
    }
  }
  console.log(arr);
}

bubbleSort([45, 2, 645, 1, 5, 6, 78, 271]);

VM55:11
▼ (8) [1, 2, 5, 6, 45, 78, 271, 645] ⓘ
  0: 1
  1: 2
  2: 5
  3: 6
  4: 45
  5: 78
  6: 271
  7: 645
  length: 8
  ► [[Prototype]]: Array(0)
```

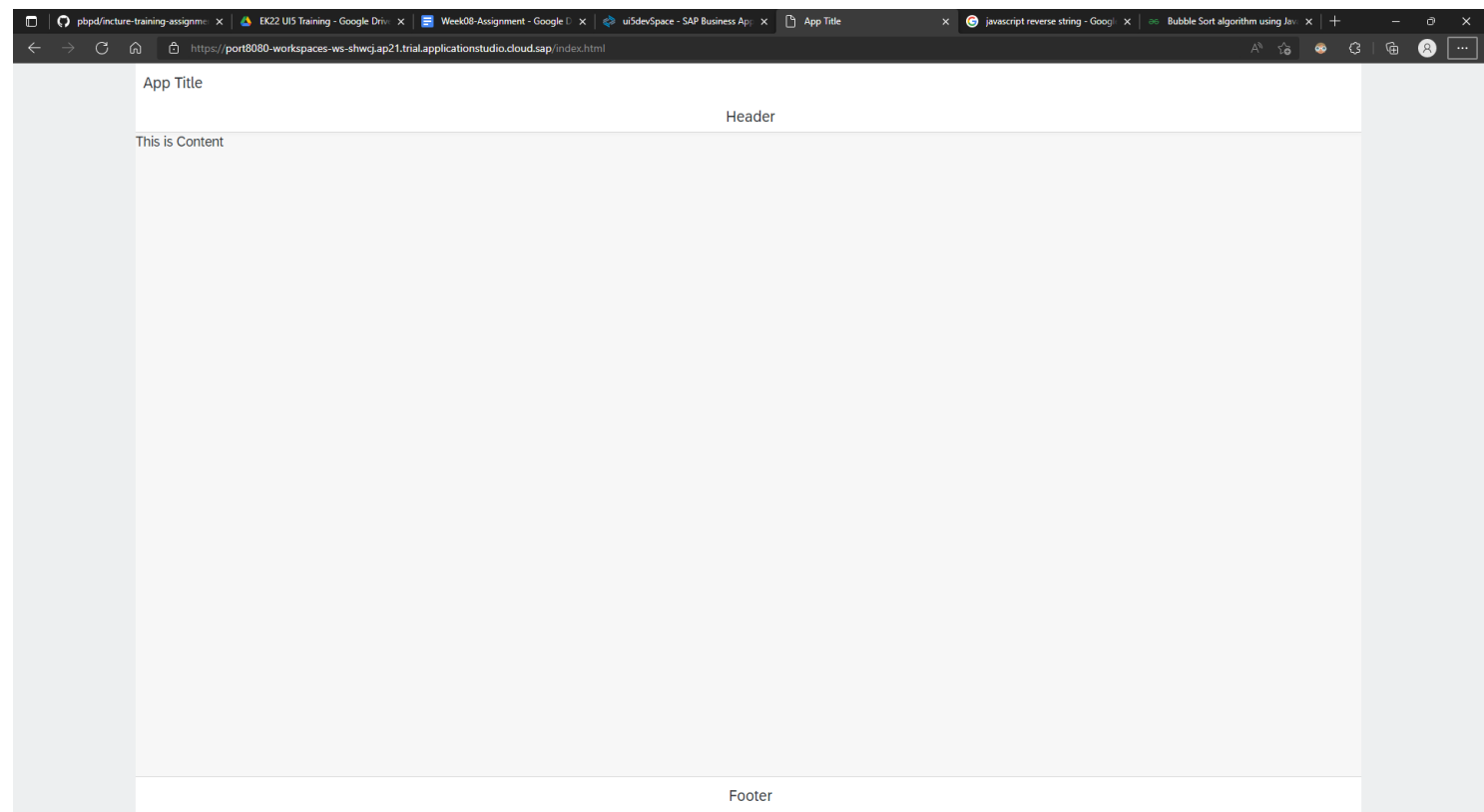
4.

Code - [q4-q6-dialog-popup \(github.com\)](#)



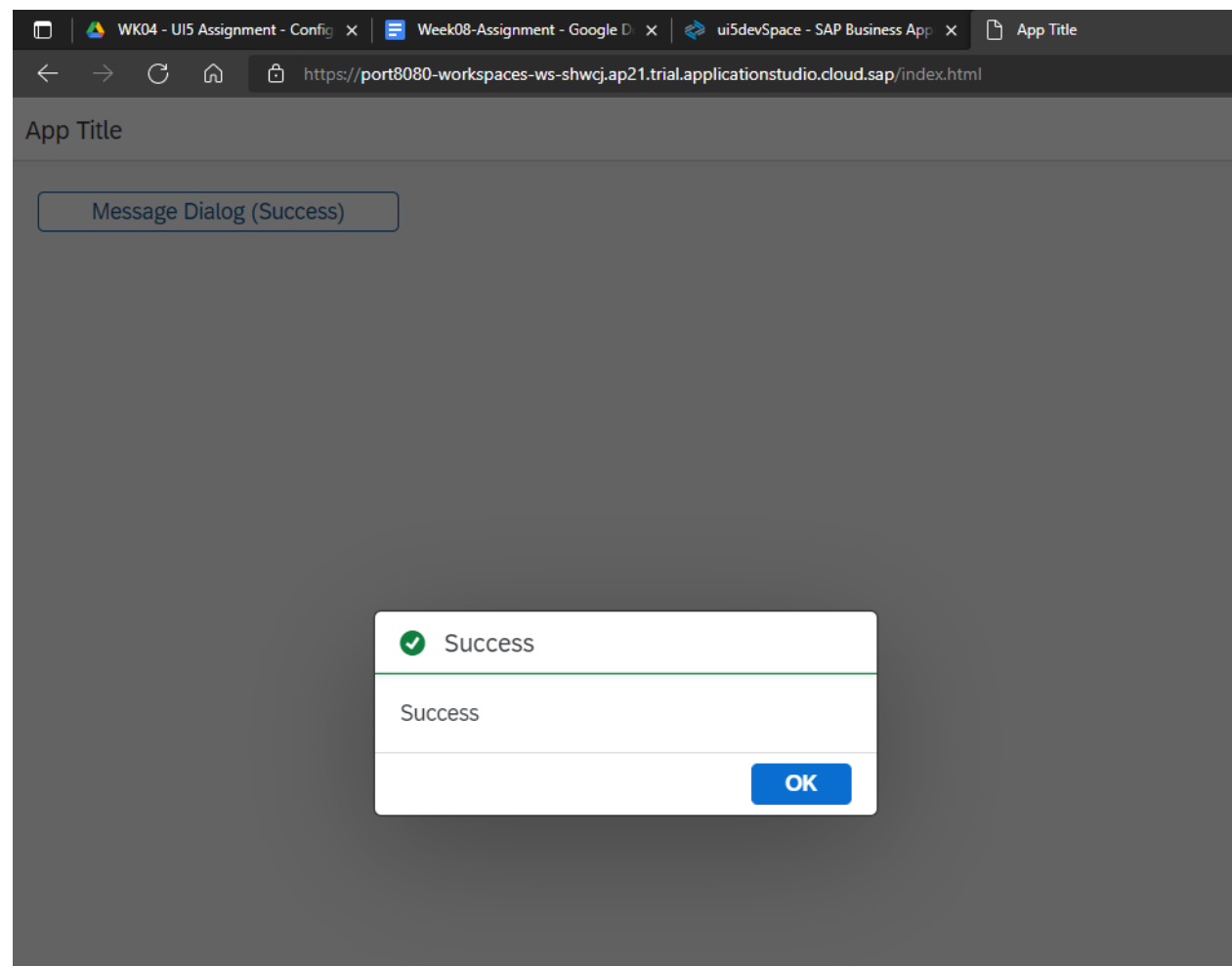
5.

Code - [q5-header-footer-fragment \(github.com\)](#)



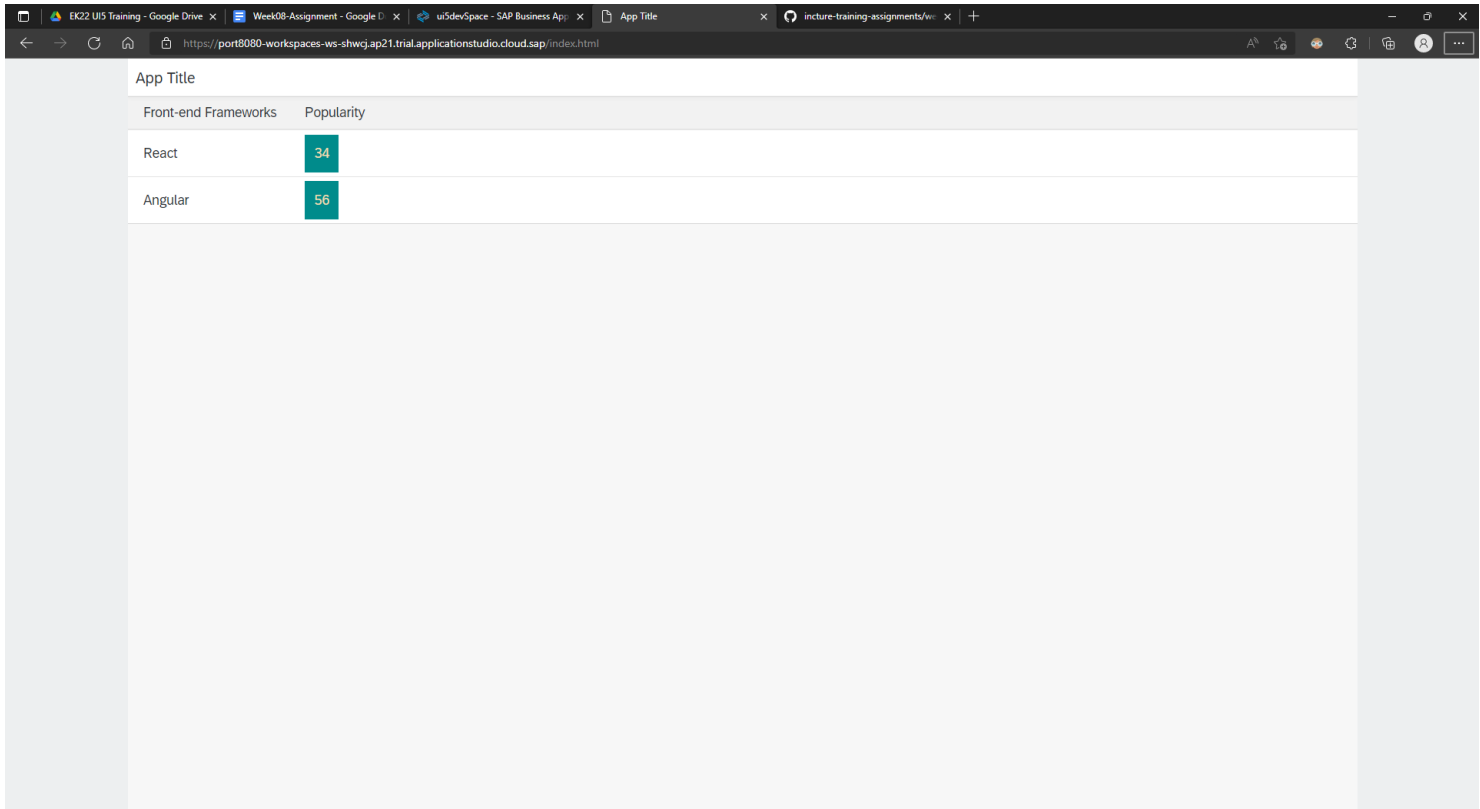
6.

Code - [q4-q6-dialog-popup \(github.com\)](#)



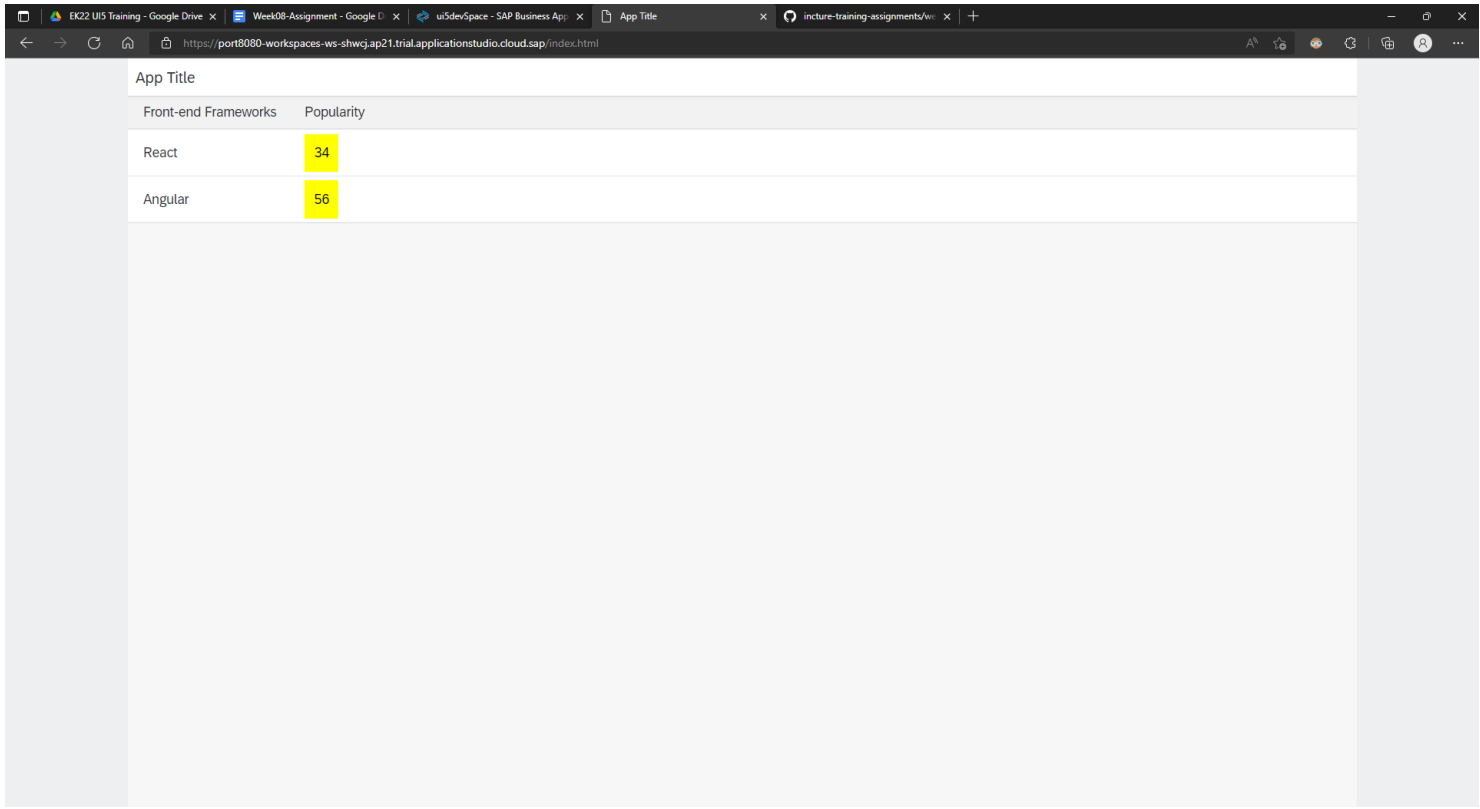
7.

Code - [q7-q8-formatter \(github.com\)](#)



8.

Code - [q7-q8-formatter \(github.com\)](#)



9. Formatter : Formatters are used to define the formatting of data on the UI while data types work in both directions: they format the data on the UI and parse and validate user input that is entered. We can either use standard formatters and data types or define your own custom objects. SAPUI5 provides standard formatter classes that can be used to define custom data types and custom formatters.

10.

Compiler stops before the formatter function.