Bachelors of Science (Hons) (MQF Level 6)

Client-side Scripting

Develop a simple web application using Angular components

Assignment Guidelines

- This assignment is in the form of a **class assignment and all tasks are required to be done in class**.

- The College operates a cheating/**plagiarism policy** and any copied work will be penalized according to this policy. **The work you present must be your own work! Plagiarism will not be tolerated!**

- Time: 2.00hrs.

What to turn in

In order for the assignment to be handed in, make sure that the following items are included:

- The assignment **cover sheet** should be fully completed with all required details together with the assignment brief. Make sure to write your name, class and **username** given on the front cover sheet.
  □
- Make sure that your lecturer has taken a copy of your work on his/her pen drive.

Brief

You are being provided with an angular template application which includes the appComponent typescript file and the appComponent html file. Images are also available under the assets folder (C:\angular).

Open the folder using Visual Studio Code. (You do **NOT** need to run npm install). Make sure to fix any errors by creating the AppComponent class, the Component decorator and importing the Component from '@angular/core'. Modify app.module.ts accordingly.

Section 1 (AA2 – 7marks)

Criteria: Use the JavaScript framework to extend HTML to contain instructions on how the model should be projected on the view.

1) Add a typescript file to the countries folder (country-list.component.ts). Create the class countryComponent and the Component decorator. Assign a value of 'app-country-list' to the selector and assign the value of 'country-list.component.html' to the templateUrl. Make use of relative pathing. [2 marks]

2) Create an array of data type any[] and assign the following 2 records: [2 marks]
"country": "Colombia",
"capital": "Bogota",
"indepDate" : "July 20, 1810",
"population": "48759958",
"flagUrl" : "./assets/images/colFlag.svg"

"country": "Brazil",
"capital": "Brasilia",
"population": "206135893",
"indepDate" : September 7, 1822,
"flagUrl" : "./assets/images/braFlag.svg"

Use Angular built-in directive *ngIf to check that the array length is not zero and that array exists. Use *ngFor to display correctly the above 2 countries on screen. Use the selector <app-country-list> to display on the app.component.html page. [3 marks]

Section 2 (AA3 – 7 marks)

Criteria: Use a JavaScript framework features to implement data-binding in a web application

Create a country model (country.model.ts) under the countries folder. The model is just a blueprint for the country objects that you will create. This class then needs to be instantiated. The class must have the following properties: country (string), capital (string), population (number), indepDate (string), flagUrl(string). [2 marks]

Add a constructor that receives the above properties. In the constructor body assign the arguments received to the properties of the object e.g. this.country = country; [2 marks]

Change the data type **any,** assigned to the countries array and use the Country model instead. Don't forget to import the Country model. [1 mark]

Section 3 (SE3 – 10 marks)

Criteria: Create filters to format the values of expressions for display to the user

1) Add a property and a method to the class AppComponent. The property should be named title and should have a string data type. Assign the value of "Country List" to the property. Create a method and name this method getTitle(). This method should return the string value of the title property. Call the method getTitle() inside your app.component.html page. Make sure that the value of property title is being shown. [2 marks]

2) Use pipes to do the following:
   a. The country name must be shown in uppercase [1 mark]
   b. Capital must be shown in lowercase [1 mark]
   c. Use the 'longDate' format with the date pipe to display the date [1 mark]
   d. Chain pipes to display the date in uppercase [1 mark]
   e. Create a button to display once again the date in 2 different formats. Once the button is clicked the date is shown in the following 2 formats: [2 marks] (Refer to appendix 2) i. 'shortDate'
      ii. 'mediumDate'

3) Create a button as shown in appendix 2 to show and hide the flag. [2 marks]

Note: All methods should be created in country-list.component.ts

Section 4 (KU6 – 5 marks)

Criteria: Select and construct a number of new features

Use property binding (**not interpolation**) to display the flag of each country (Refer to appendix 2). [1 mark]

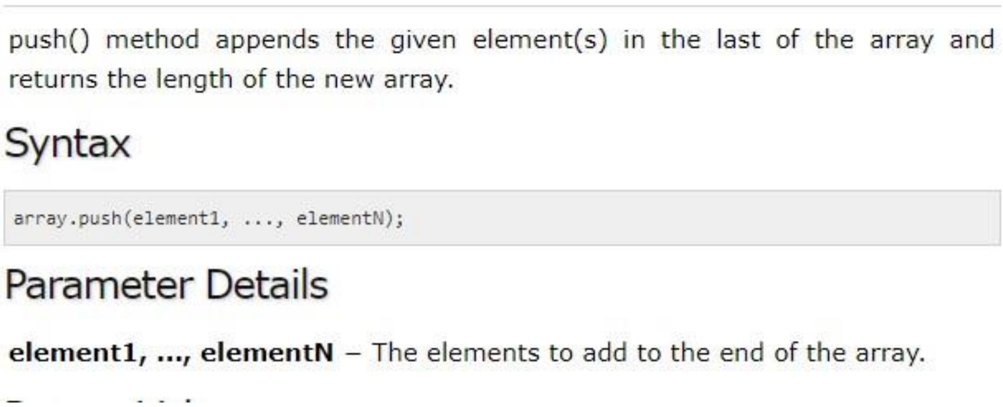
Create 2 new buttons as shown in Appendix 3. One should be used to call a method to add a new country and another one to delete the last country of the list.

To add an object country to the list use the push method. Just pass as an argument a new object.

Use the new keyword and details as shown below to create the country object. [2 marks]

```
(new Country("USA", "Washington DC", "July 4, 1776", 323947000,
"./assets/images/usaFlag.svg"))
```

Note: All methods should be created in country-list.component.ts



push() method appends the given element(s) in the last of the array and returns the length of the new array.

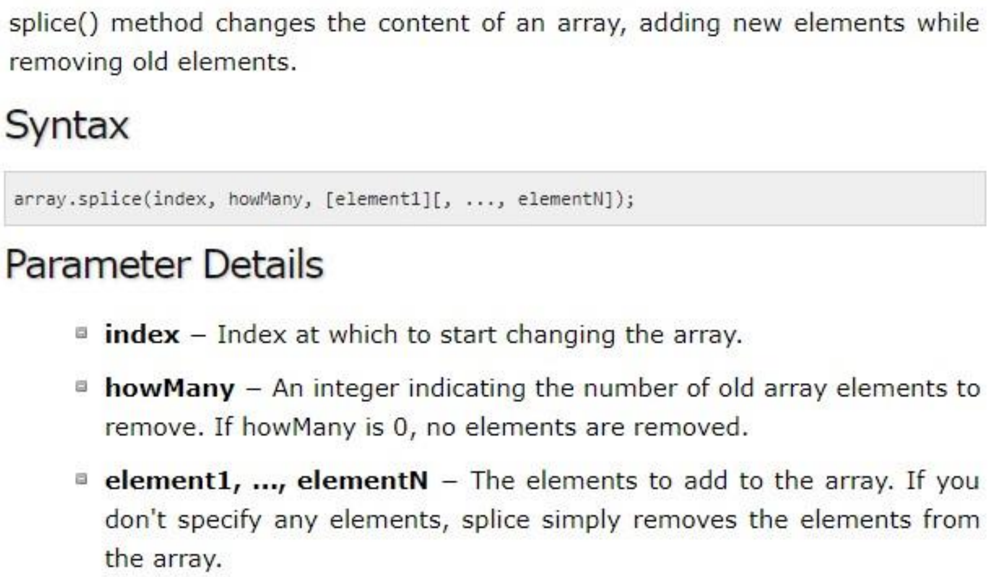
Syntax

```
array.push(element1, ..., elementN);
```

Parameter Details

element1, ..., elementN – The elements to add to the end of the array.

To delete the last country from the list use the splice method. Use this.countries.length-1 to start from the last element.



splice() method changes the content of an array, adding new elements while removing old elements.

Syntax

```
array.splice(index, howMany, [element1][, ..., elementN]);
```

Parameter Details

- **index** – Index at which to start changing the array.
- **howMany** – An integer indicating the number of old array elements to remove. If howMany is 0, no elements are removed.
- **element1, ..., elementN** – The elements to add to the array. If you don't specify any elements, splice simply removes the elements from the array.

Once you have completed the solution, remove the node_modules folder and submit the zipped solution to your lecturer.

--END OF ASSIGNMENT--

MARKING SCHEME

AA2	**Use the JS Framework to extend HTML to contain instructions on how the model should be projected on the view**	Max	Awarded
	1) Creating the component countries correctly (class, decorator and html file) and make use of relative pathing.	2	
	2) Creating the array with data type any[] and populating correctly.	2	
	3) Using the *ngIf and *ngFor properly. [1 mark] Check that the array length is not zero. [1 mark]. Check that the array exists and if not table header is not shown. [1 mark]	3	
Total		**7**	

AA3	**Use a JavaScript framework features to implement databinding in a web application**	Max	Awarded
	Create correctly a country model (country.model.ts) under the countries folder.	2	
	Add a constructor that receives all properties. In the constructor body assign the arguments received to the properties of the object e.g. this.country = country;	3	
	Change the data type **any,** assigned to the countries array and use the Country model instead.	2	

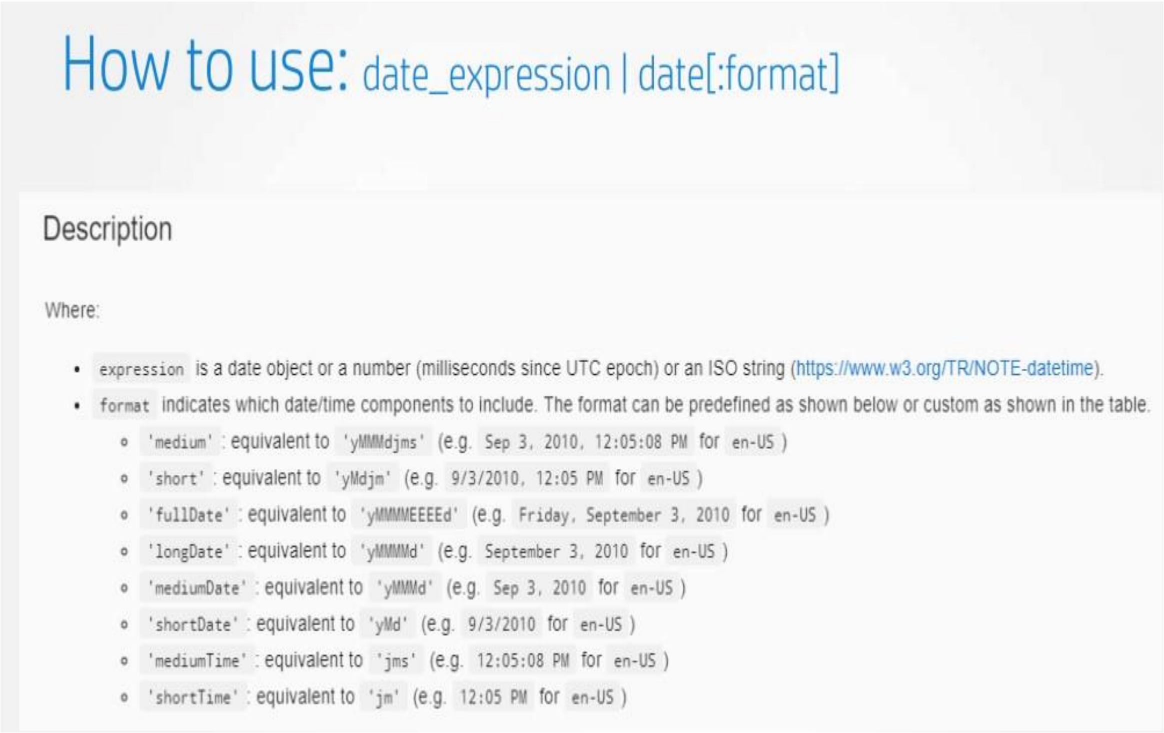
Total		**7**	

SE3	**Create filters to format the values of expressions for display to the user.**	**Max**	**Awarded**
	1) Add a property and a method to the class AppComponent. The property should be named title and should have a string data type. Assign the value of “First Practical” to the property. Create a method and name this method getTitle(). This method should return the string value of the title property. 2) Use pipes to do the following: a. The country name must be shown in uppercase b. Capital city must be shown in lowercase c. Use the ‘longDate’ format with the date pipe to display the date d. Chain pipes to display the date in uppercase e. Create a button to display once again the date in 2 different formats. Once the button is clicked the date is shown in the following 2 formats: [3 marks] i. ‘shortDate’ ii. ‘mediumDate’ 3) Create a button as shown in appendix 2 to show and hide the image.	2 1 1 1 1 2 2	
Total		**10**	

KU6	Select and construct a number of new features	**Max**	**Awarded**

	Displaying image properly using property binding Create 2 buttons. One should be used to add a new country [2 marks] and another one to delete the last country of the list [2 marks]. Refer to appendix 3	1 4	
Total		**5**	

APPENDIX 1



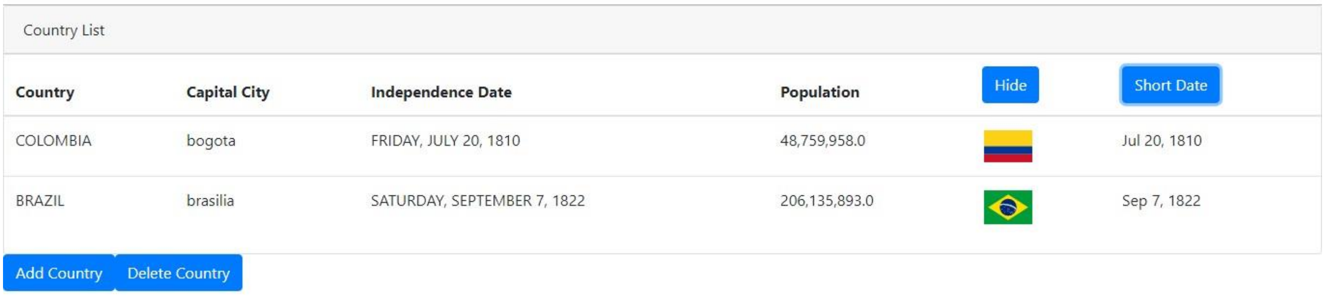
How to use: date_expression | date[:format]

Description

Where:

- `expression` is a date object or a number (milliseconds since UTC epoch) or an ISO string (https://www.w3.org/TR/NOTE-datetime).
- `format` indicates which date/time components to include. The format can be predefined as shown below or custom as shown in the table.
  - `'medium'` : equivalent to `'yMMMdjms'` (e.g. `Sep 3, 2010, 12:05:08 PM` for `en-US`)
  - `'short'` : equivalent to `'yMdjm'` (e.g. `9/3/2010, 12:05 PM` for `en-US`)
  - `'fullDate'` : equivalent to `'yMMMMEEEEd'` (e.g. `Friday, September 3, 2010` for `en-US`)
  - `'longDate'` : equivalent to `'yMMMMd'` (e.g. `September 3, 2010` for `en-US`)
  - `'mediumDate'` : equivalent to `'yMMMd'` (e.g. `Sep 3, 2010` for `en-US`)
  - `'shortDate'` : equivalent to `'yMd'` (e.g. `9/3/2010` for `en-US`)
  - `'mediumTime'` : equivalent to `'jms'` (e.g. `12:05:08 PM` for `en-US`)
  - `'shortTime'` : equivalent to `'jm'` (e.g. `12:05 PM` for `en-US`)

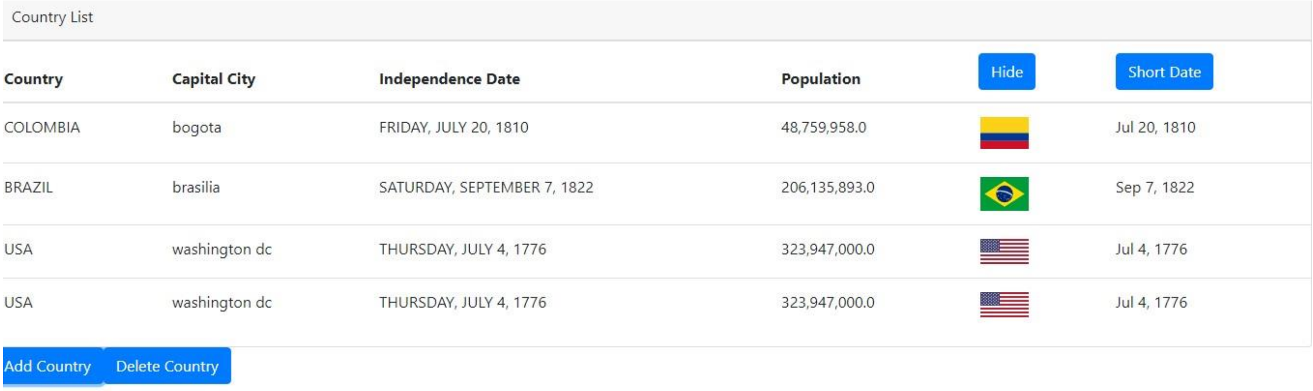
APPENDIX 2



Country List

Country	Capital City	Independence Date	Population	Hide	Short Date
COLOMBIA	bogota	FRIDAY, JULY 20, 1810	48,759,958.0	🇨🇴	Jul 20, 1810
BRAZIL	brasilia	SATURDAY, SEPTEMBER 7, 1822	206,135,893.0	🇧🇷	Sep 7, 1822

APPENDIX 3



Country List

Country	Capital City	Independence Date	Population	Hide	Short Date
COLOMBIA	bogota	FRIDAY, JULY 20, 1810	48,759,958.0	🇨🇴	Jul 20, 1810
BRAZIL	brasilia	SATURDAY, SEPTEMBER 7, 1822	206,135,893.0	🇧🇷	Sep 7, 1822
USA	washington dc	THURSDAY, JULY 4, 1776	323,947,000.0	🇺🇸	Jul 4, 1776
USA	washington dc	THURSDAY, JULY 4, 1776	323,947,000.0	🇺🇸	Jul 4, 1776