Lab 3: React Native Practice

Note:

- For the activity 1-3, you can directly create JavaScript file in your VS Code environment.
- For Activity 4 and 5, you may can go for the below setup
  - These steps will be required to create your RN app (Create a React Native CLI application after following the instructions from [here.](#) Run the application on your android/IOs environment.)

Activity 1:
1. Get used to the editor, run first JS program.

2. Write biography about yourself and print on console.

3. use 'var' to store your biography in variables, use appropriate primitive types.

4. Create JS Object for your biography key-value pairs. At least 4-5 keys, nested JS Object for Address, DegreePrograms etc

e.g. `{ name: 'Some Name', age: 34, address: { }, degreeProgram: {} }`
Print biography on console (do not print entire object as it is)

Activity 2:
Find the sum of all the multiples of x or y below z.
If we list all the natural numbers below 10 that are multiples of 3 or 5, we get 3, 5, 6 and 9. The sum of these multiples is 23. (3+5+6+9). Find the sum of all the multiples of x or y below z.
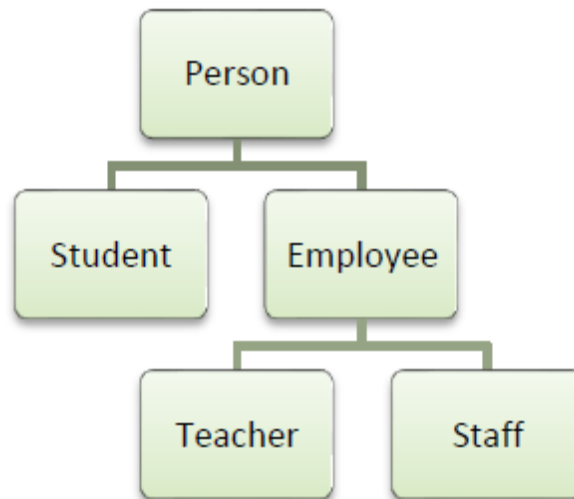
Activity 3:
Implement min and max methods which returns minimum and maximum value of supplied arguments. Implement your own algorithm to find the minimum and maximum value.
- min(4,8,1,3) // returns 1
- max(4,6,5,3,2) // returns 6

Activity 4:

Implement the following scenario: Student Management System containing Person, Student, Employee, Teacher, Staff and Courses functions.

- Student and Employee functions are the derived class of Person
- Teacher and Staff are derived functions of Employee
- Person class contains 3-4 common fields of Student and Employee. Initialize the fields with the default value
- Employee class contains 3-4 common fields of Teacher and Staff (such as, department, designation, salary etc)
- Student, Teacher and Staff contains the fields specific to their role
- Courses has aggregation type of association with Student and Teacher functions

Finally, create two objects of Student, Teacher and Staff each and print their information through created objects on console.

Activity 5:

The goal of this lab is to practice React Native, starting with the code from lesson 5. In it, we defined a list of users, that we are displaying. Of note, these users are the same as the ones you will use for the assignment, so this lab will also be useful to familiarize yourself with the assignment.

You can extend this code in the following ways:

- Change the style of the components to increase the spacing between users, and the padding around their profile pictures. Display the last name in bold font (but not the title or the first name).
- Display each user's phone numbers and email address, being mindful of the style of the application. For this, you can consult the layout documentation.
- Change the background of each user to reflect a different color for each country (you can only take a subset of the countries; you don't need a color for all).
- Highlight the users that have a weak password (a weak password being a password of length less than 8 characters).
- Define an enhanced view of each user, showing more data, such as their city, their country, their phone number, and a larger picture. For this, you can consult the layout documentation.

- Define a "full screen" view of each user, showing as much detail as possible in one screen. Use the "select from list" idiom so that tapping one of the users will show it, also allowing for the user to be deselected.

- Add a text field on top of the application to implement a search box. Entering text filters the list of displayed users so that they match the search string.

- Change the behavior of the list so that it changes the way users are rendered depending on how many search results there are. Many users will have a compact view, few will have an expanded view, and a single result switches to the full screen view.

You can edit the starter code at this link