Course Authored By:
```
------------------------------------------------------------------------
----------------------------------
```
Srinivasan Kannan
(also known as: Shrinivaasan Kannan, Shrinivas Kannan)
Ph: 9791499106, 9003082186
Krishna iResearch Open Source Products Profiles:
http://sourceforge.net/users/ka_shrinivaasan,
https://github.com/shrinivaasanka,
https://www.openhub.net/accounts/ka_shrinivaasan
Personal website(research): https://sites.google.com/site/kuja27/
emails: ka.shrinivaasan@gmail.com, shrinivas.kannan@gmail.com,
kashrinivaasan@live.com
```
------------------------------------------------------------------------
----------------------------------
######################################################################
##################################################################
```

This is a non-linearly organized, code puzzles oriented, continually
updated set of course notes on AngularJS. This
complements NeuronRain course materials on Linux Kernel, Cloud, BigData
Analytics and Machine Learning.
```
------------------------------------------------------------------------
------------------------------------------------------
```

2 June 2017

AngularJS is the client side Advanced Java Scripting (AJAX) engine
runtime supported by most browsers. AngularJS follows Model-View-
Controller pattern:
    - View is the HTML visible to user
    - Model is the form layout and frames of the Dynamic HTML page
    - Controller acts on the model data and eschews HTML output from
Server Side JavaScript (Node.js) or other
frameworks(Django/Flask/Tornado)

Model-View-Controller decouples the HTML rendering from the hidden
dynamic logic. NeuronRain GUI frontend has two client/server interfaces:
Plain HTML-to-Tornado and AngularJS-to-Tornado webservers
(https://github.com/shrinivaasanka/asfer-github-code/tree/master/python-
src/webserver_rest_ui/). AngularJS HTML template for a simple form submit
has following structure:
    - form tag with ng-app directive defining the application name of
the controller
    - Model name for input elements in the form
    - script tag which defines a controller object from ng-app and
defines a function for form processing invoked from submit widget of the
form

- AngularJS script in controller has access to builtin(s) like $scope, $http etc., for accessing model data in the scope. Form data can be submitted to a REST URL with GET or POST with stringified JSON data.