**Angular JS – Day 1**

**(Directives and Data Bindings)**

**Exercises**

1. Declare a textfield as model

Bind this to <h2> as view.

The style for h2 should be italics and red colour

2. Declare two text fields as data models

Bind them to two views h1 and h2

Style for h1 should be red background and white text

Style for h2 should be green background and black text

3. Create a text field for accepting the city name.

Set this model to h1 and h2.

Objective:

Setting the model in more than one views h1 and h2

4.How to initialize a variable in Angular JS?

Intialize a variable mycountry to "India"

Create TextField for accepting the country name

Display the country name in <h2> tag with the text i.e. set the view to h2.

I live in \_\_\_\_\_\_\_\_ (country name as typed by the user)

Till the user enters the new country name, the value “India” should be displayed in the textfield.

5. Initialize two variables one with your name and your batch name.

Display as follows, set the view to P tag using expression.

Guru is in the batch ALPHA5555

6. Initialize two numeric variables with values 10 and 20

Display their product (10\*20).

7. Create Product AngularJS object with properties productName and productPrice and display them in Angular JS

a. using Angular expressions

b. using ng-bind directives

8. Create an AngularJS array with the country names:

China, India, USA, Germany, Australia

Display the following using array index and using Angular expressions.

China is our neighbouring country

India is our mother land

USA is the leading country in IT

Germany is famous for automobiles

World cup 2015 was held in Australia

9. Create a Sales object:

Property names:

product name

product price

Assign some values to these properties:

Example:

HP LAPTOP

50000.00

Using Angular expression, set the view to P tag to display the following.

HP LAPTOP is sold for Rs. 50000.00

10. Create two textfields:

One for accepting name

One for accepting city

Bind these two models into a single view <h1>

Objective:

Binding more than one model into single view