Oefenopdracht 1

2 way binding: Wanneer er iets veranderd in het model, dan kan je dat gelijk zien. Wanneer er iets veranderd *in the view* dan wordt de model gelijk geupdate.

Oefenopdracht 2

AngularJS maakt gebruik van een model-view-controller. Dit is heel handig om dynamische websites te maken.

Oefenopdracht 3

Wanneer er te veel variabelen zijn in de $scope die moeten worden gecheckt. “This approach potentially leads to slow rendering when AngularJS checks on too many variables in the $scope every cycle. “

Studieblok 2

Oefenopdracht 1

NG-app = wordt gezien als de root element van de gehele applicatie.

NG-model = dit verbind de value van een html element aan de applicatie.

NG-bind = vervangt de inhoud van het html element met de value van het gegeven variabele

Studieblok3

Oefenopdracht 1

De 1 is tussen 2 paar curly brackets {{ expression }} en de ander in een directive ng-bind = “expression”.

Studieblok 4

Oefenopdracht 2

Je kan namelijk angular.module pas gebruiken als de bibliotheek is geladen. (Dat kan dus niet onderaan je code).

Oefenopdracht 5???!!!

Je kan beter geen bestaande elementen gebruiken omdat die elementen daarvoor niet zijn bedoeld.??

Het is handig om “van buiten af” informatie in een element te kunnen stoppen