ng-csp DIRECTIVE IN ANGULARJS

- The **ng-csp directive** in AngularJS used to change the CSP **(Content Security Policy)** rules.

- If the **ng-csp** directive is set into the HTML element, then the AngularJS will not run any **eval functions** and **inject any inline styles**.

- ng-csp directive is needed when developing apps for google chrome extensions or windows apps.

- Some of the following rules affect the AngularJS:

  o **Use of eval():** function(string) and similar functions are dynamically creating and execute code from strings is not allowed in the eval () function. Use $parse service in AngularJS to provide 30% increase in the speed of the evaluating expression.

  o **Use of Inline style**: Inline resources such as <script> and <style> elements are not allowed. It prevents the apps from injecting custom styles directly into the document. AngularJs include the CSS rules in **ngCloak** and **ngHide** directive. These directive work when a CSP rule is blocking inline styles and the link must be provided to the **angular-csp.css** in HTML Manually.

- This directive is only available in the ng-csp and data-ng-csp attribute form.

no-inline-style:

ng-csp directive value is set to no-inline style, it will stop AngularJS from injecting any CSS inline styles in the DOM, but allow eval function.

Syntax for no-inline-style in AngularJS:

```
<element ng-csp="no-inline-style"></element>
```

Parameter Value:

Value	Description
no-inline-style	States that the value can be one of the two values described. If the value can be both values, separated by a semicolon, it means empty value.

no-unsafe-eval:

ng-csp directive value is set to no-unsafe-eval function, it will stop AngularJS from running any eval functions, but allow injecting inline styles.

Syntax for no-unsafe-eval in AngularJS:

```
<element ng-csp="no-unsafe-eval"></element>
```

Parameter Value:

Value	Description
no-unsafe-eval	The value can be empty, means eval or inline styles are not allowed.

No Declaration of ng-csp Directive:

No declaration means that angular will assume to it can use inline styles and it will do a runtime check for unsafe-eval.

Syntax

```
<body >
```

Deactivate in-line-style and unsafe-eval:

The ng-csp and data-ng-csp attribute in AngularJS to deactivate both inline styles and unsafe eval.

Syntax

```
<body ng-csp >
```

Both no-unsafe-eval and no-inline-style:

ng-csp directive is set to both value is no-unsafe-eval and no-inline-style in angularJS it means that does not inject styles nor use eval functions, which is same as an empty.

Syntax

```
<body ng-csp="no-inline-style:no-unsafe-eval">
```