***!! Shri Ganeshay Namah: !!***

HTML Coding Standards

# **ID Attribute**

1. **Uniqueness**

* **Ensure Uniqueness:** The Id Attribute Must Be Unique Within The HTML Document. This Is Crucial Because Many Client-Side Scripts Rely On The Id To Locate Elements.

1. **Naming Conventions**

* **Descriptive Names:** Use Meaningful Names For Id Attributes That Describe The Element's Purpose Or Content. This Improves Readability And Maintainability.
* **Small Case:** Follow Consistent Naming Conventions, Like small-case (E.G., first-name-input, person-age-input) To Maintain Uniformity.
* **Prefix:** A Prefix Denoting The Type Of Object Is Considered Good Practice When Naming Accordion / Any Block (E.G. Accordion, Data Tables). The Suggested Way To Use These Prefixes Is To Concatenate Them With The Accordions Or Data Table Name And An Additional Element Denoting The Function The Constraint Performs.
* **Avoid Redundancy:** Another Common (But Discouraged!) Practice Is To Prefix Each Id Name With An Abbreviation Of The Accordion / Block Name. This Adds Unnecessary Redundancy And Makes So Long And Difficult To Read And Write.

1. **Avoiding Collisions**

* **Partial Views And Components:** When Using Partial Views Or Reusable Components, Ensure That The Id Attributes Are Unique Across The Entire Page. You May Need To Prefix Or Suffix Ids With The Component Name Or Context To Avoid Conflicts.
* **Data Binding:** When Using Loops (E.G., foreach) To Render Lists Of Elements, Ensure That The Id Attributes Are Unique, Possibly By Appending The Loop Index Or A Unique Identifier To The Id.

1. **Accessibility**

* **Label Association**: When Creating Forms, Ensure That Label Elements Are Associated With Their Corresponding Inputs Using The For Attribute, Which Should Match The Id Of The Input Element. This Improves Accessibility. (In Our Example – Radio Button)
* **ARIA Attributes:** Use Id Attributes In Combination With ARIA (Accessible Rich Internet Applications) Roles And Properties To Improve The Accessibility Of Complex UI Components.

1. **Client Side Interactions**

* **JavaScript And JQuery**: When Using Javascript Or Jquery To Manipulate DOM Elements, The Id Attribute Is Often Used As A Selector. Be Cautious Of Hardcoding Id Values Directly In Scripts; Consider Passing Them Dynamically To Avoid Maintenance Issues.

1. **Rendering Complex Controls**

* **Custom Helpers**: If You Create Custom HTML Helpers, Ensure That They Generate Id Attributes That Follow The Same Conventions And Avoid Duplications.
* **Dynamic Elements:** For Elements Generated Dynamically Via AJAX Or Other Client-Side Techniques, Ensure That The Id Attributes Remain Unique And Follow The Conventions Used In The Rest Of The View.

**Conclusion**

By Following These Standards, You Can Create Well-Structured, Maintainable, And Accessible MVC Views.