STANDARD CODING CONVENTIONS

C-TALK

Version 1.1

Author:
KARRA SANDEEP(201452037)
Sai Prahlad (201452052)

November 11, 2016

Indian Institute of Information Technology Vadodara

Team members :

TEAM MEMBER	ID
Bhoopendra Singh	201452020
Shikhar Dhing	201452021
Venkata Sandeep	201452037
Anjali Kumari	201452042
Vipin Sahu	201452051
Prahlad	201452052
Sachin Jangid	201452060
Sunny Sankhlecha	201452061
Kenneth Tenny	201452066

Revision History

Version	Description	Authors	Reviewers
1.1	CODING CONVENTION	KARRA SANDEEP, SAI PRAHLAD	SHIKAR DHING, BHOOPENDRA SINGH

Date	Version no.	Authors
15/11/2016	1.0	Karra venkata sandeep

Table of Contents

1. Introduction

1.1. **Purpose:**

The main purpose of this document is to create coding habit amongst project group members. By maintaining proper coding standards, the code written by different group members can be easily understood. Reviewing and maintaining of code becomes easier.

If we follow proper coding standard we can be benefited in the following ways:

1. Any programmer (group member) will be able to read and understand any part of the code.
2. Maintainability and reviewing becomes lot easier.
3. If a member is new to coding they can quickly become accustomed to following the existing standards.
4. A programmer new to the language is spared making the same mistakes over and over again, so reliability is increased.
5. It becomes lot easier to make corrections if any bugs are found.

1.2. **Scope:**

This document describes general coding standards followed in writing any code. Coding convention of all the languages used are to be described in this document. Also proper guidelines are to be followed for each language.

1.3. **Coding standard documents:**

Each project has a particular set of coding standards. Coding standards consist of following parts:

- General coding standards must be mention in this document.
- Specifying coding standards for each languages used.
- Language standards won’t affect general coding standards. It will rather help.
- Project coding standards are based on coding standards in this document and coding standard of languages.
- The project standard should be such that there is no conflict with language standards and general standards.
- If there is any kind of conflict then project standards are to be considered.

2. File and Module Guideline

2.1. File Suffixes

We will use following suffixes:

File Type	**Suffix**
JS File	.JS
HTML File	.html
CSS File	.css
JSON File	.json

2.2. File Organization:

In the file, if there are sections then each sections should be separated by blank lines or comments about than section.

2.3. Source files:

We have followed standard coding conventions of

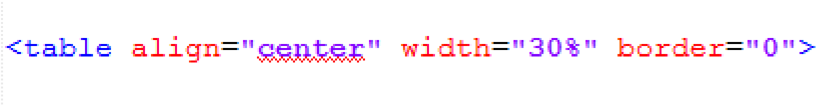
2.3.1. HTML Source Files:

1). **Self-closing Elements:**

All tags must be properly closed. For those tags that are self-closing, the forward slash should have exactly one space preceding it. W3C standards specifies that single space should precede the self-closing slash.

2). **Attributes and Tags:**

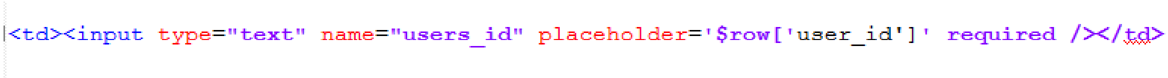
All tags and attributes are in lowercase. As purpose of text is to be interpreted by machines therefore attribute value is lowercase. Data needs to be human readable and also proper title capitalization should be followed.



```
<table align="center" width="30%" border="0">
```

3). **Quotes:**

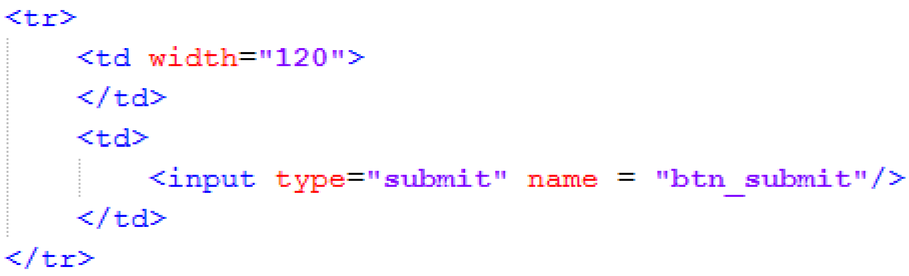
W3C standard for HTML5 implies that all attributes must have a value, and must use double or single quotes. Following are example of quotes:



```
<td><input type="text" name="users_id" placeholder='$row['user_id']' required /></td>
```

4). **Indentation:**

We have used tabs instead of spaces for indentions.



```
<tr>
    <td width="120">
    </td>
    <td>
        <input type="submit" name = "btn_submit"/>
    </td>
</tr>
```

5). **Naming Convention:**

Abbreviate words when needed. i.e. Button – btn. Separating words by using underscores or hyphens. For mentioning action initially letter must be in uppercase.

2.3.2. JavaScript Source Files:

1). **Braces:**

For multiline blocks, braces should be mention in same line.

2). **Whitespace:**

We used CamelCase for naming variables and functions

3). **File Name:**

There are various JS files separated in models, views, controllers folder.

4). **var keyword:**

Var statement is used to declare any local variable necessary. Each variable is separated by a single comma. The variable or variables defined by var statement should be mention on separate line.

5). **Loops:**

There are many loops used in JavaScript. We have used for loop in most part except while iterating JSON objects.

2.3.4. CSS Source Files:

For CSS files we have used standard conventions of 1 tab indentation, bracket in next line,etc.

3. References

Standards and Coding Conventions v1.0, Team 2, IT-314 Software Engineering, Winter 2016-17.