



INDIAN INSTITUTE OF INFORMATION TECHNOLOGY, VADODARA

MEETUP

Coding Standard

Author:

POOJA GURJAR

Date: 20-10-2018

Contents

1	Introduction	2
1.1	Purpose	2
1.2	Scope	2
2	Django Coding Style	2
2.1	Python Style	2
2.2	Imports	3
2.3	Template Style	3
2.4	View Style	3
2.5	Model style	3
3	HTML and CSS Coding Style	3
3.1	Indentation	3
3.2	Capitalization	4
3.3	Trailing Whitespace	4

1 Introduction

1.1 Purpose

A coding standards document tells developers how they must write their code. Instead of each developer coding in their own preferred style, they will write all code to the standards outlined in the document. This makes sure that a large project is coded in a consistent style - parts are not written differently by different programmers. Not only does this solution make the code easier to understand, it also ensures that any developer who looks at the code will know what to expect throughout the entire application. The goal of this document is to create uniform coding habits among software personnel in the project team so that reading, checking, and maintaining code written by different persons becomes easier. The intent of these standards is to define a natural style and consistency, yet leave to the authors of the project source code, the freedom to practice their craft without unnecessary burden.

1.2 Scope

This document describes general software coding standards for code written in python, HTML and CSS. Since, these will be the languages which we will be using to develop the website. For each project, this document will be used in conjunction with language and define a complete set of coding standards.

2 Django Coding Style

2.1 Python Style

- Use four spaces for indentation.
- Use single quotes for strings, or a double quote if the string contains a single quote. Don't waste time doing unrelated refactoring of existing code to conform to this style.
- Use underscores, not camelCase, for variable, function and method names.

2.2 Imports

- Put imports in these groups: future, standard library, third-party libraries, other Django components, local Django component, try/excepts. Sort lines in each group alphabetically by the full module name. Place all import module statements before from module import objects in each section. Use absolute imports for other Django components and relative imports for local components.
- On each line, alphabetize the items with the uppercase items grouped before the lowercase items.

2.3 Template Style

- In Django template code, put one (and only one) space between the curly brackets and the tag contents.

2.4 View Style

- In Django views, the first parameter in a view function should be called Request.

2.5 Model style

- Field names should be all lowercase, using underscores instead of camel-Case.
- The class Meta should appear after the fields are defined, with a single blank line separating the fields and the class definition.

3 HTML and CSS Coding Style

3.1 Indentation

- Indent by 2 spaces at a time.

3.2 Capitalization

- Use only lowercase. All code has to be lowercase: This applies to HTML element names, attributes, attribute values (unless text/CDATA), CSS selectors, properties, and property values (with the exception of strings).

3.3 Trailing Whitespace

- Remove trailing white spaces. Trailing white spaces are unnecessary and can complicate diffs.