



**Hiranya Garbha
Consultancy**

Incubating Golden Ideas

PROGRAMMING GUIDELINES

Sanjay G
sanjay@hiranya-garbha.com

Programming Guidelines

Defining functions:

Use a lowercase word or words. Separate words by underscores to improve readability.

Example: 1) add_limit_item()

2) generate_guid()

Defining class:

Start each word with a capital letter. Do not separate words with underscores. This style is called camel case.

Example: 1) class CreateLimitForm

2) class CartItemDetails

Declaring Variables:

Use a lowercase single letter, word, or words. Separate words with underscores to improve readability.

Example: 1) header_name

2) item_name

Note: **Do not use numbers while naming variable**

Example: 1) item1

2) variable1



Creating a new App:

- App name: While creating a new app in a project always suffix with 'eProc_'

Example: 1) eProc_Shopping_Cart

2) eProc_Basic

- Create a new directory named “Utilities” and create two python files for storing application specific utility programmes and application generic utility programmes

Example: Utilities/ 1) <app_name>_specific.py 2)

<app_name>_generic.py

- Create a new directory named “templates” to store app specific templates and create a sub-folder in it to store templates so that it avoids naming conflicts if any.

Example: 1) templates / Shopping_Cart / sc_first_step.html

2) templates / Application_Settings / number_ranges.html

- Create a new directory to store forms for application

Example: 1) Shopping_Cart_Forms / limit_form.py

2) Login_Forms / login_form.py



Constants and Messages:

Create a constant for maintaining recurring information, like displaying success messages or error messages.

Example: Message

MSG001 = 'Item added to cart successfully'

Constant:

CONST_ACC_CAT = 'ACC_CAT'

Adding Comments :

Make sure to add comments above the code after completing a functionality

Example: 1) # This is single line comment

2) """

This is multiple

Line comment

"""



General coding guidelines:

- Make sure to remove temporary variables after developing a new functionality or modifying an existing one.
- Remove unwanted imports and packages that are not being used in the project.
- If installed with a new package, make sure to add the package name in project's requirements.txt
- Add a doc_string to each function or a method that has been defined and describe the parameters defined
- Add a doc_string at the top of the .py file to define all the functions and classes used in that file and describe the use of class or function
- Provide a proper indentation after defining a new function or implementing if statements or for-loop's
- Do not use a whitespace inside parenthesis, avoid redundant spaces.
 - *Example:*
 - 1) (a == b) (*wrong*)
 - 2) (a==b) (*correct*)
- Avoid redundant or unwanted parenthesis:
 - *Example:*
 - 1) if(is_valid==True): (*wrong*)
 - 2) if is_valid: (*correct*)

