

## **1.What is PEP 8 and why is it important?**

**Ans:** PEP 8 is also known as Zen of python, because it is a set of guideline principles for writing the code in python. Pep 8 is one of the best style guideline for writing code in python. It outlines a set of conventions and guidelines that developer should follow to make the python programs consistent and readable. PEP stands for Python Enhancement Proposal, and PEP 8 specially focusses on the styling conventions for writing the code in python. It covers all the aspects such as how to write comments, naming conventions, layout of code, indentation, documentation, tool support etc. The aim of PEP 8 is to make python programs easily understandable and easily accessible to python developers.

### **Variable convention:**

It specifies the guidelines how we should name our variables, functions, classes and modules so that by reading the python code the programmer should be able differentiate among them e.g. Python's class name should start with the capital letter and function name or variable names should start with smaller letters and if name contains more than one word then separate them with help of underscore(follow snake\_case).

e.g. class name: **ComplexNumbers**

function or variable name : **complex\_numbers**

If same convention is followed through writing whole python program, makes it consistent and developers can understand the purpose and usage of different elements present inside the code.

This is helpful when multiple users are making contribution in the same project.

### **Layout of Code:**

Pep 8 also provides guidelines for how our code layout should be while writing the python programs e.g. indentation (how many spaces), length of line, spacing around the operators, and when to leave blank lines.

It improves the structure of our python programs.

### **Helps in Documentation:**

PEP 8 encourages the developers how we should write the proper comments that are best to explain the purpose of each line, function or block of statements written inside the python program. In this way proper commenting above the code line, function or classes helps in documentation of the program and makes it easier to explain and understand.

PEP 8 is best practice mostly when multiple contributors and contributing on the same project, allows code consistency, readability proper layout of the code.

## **2. What is an SVM classifier? Explain by taking an example.(300 Words)**

**Ans.:** **SVM** stands for Support Vector Machine, it is one of the supervised machine learning technique which we can use for performing both regression and classification tasks. However, it is most commonly used for performing the classification tasks. SVM is also named as Large Margin Classifier. It is named as larger margin classifier because only that decision-boundary gets selected which gets the maximum margin . It is named as SVM because we use two support vectors for finding the best hyperplane. These two support vectors are the points closer to the decision boundary along positive and negative hyperplane. It is one of the powerful technique as it can be used to solve even complex non-linear functions and can segregate the data points in the higher dimensional space

There are two types of SVM's

- a) Hard-Margin Classifier.
- b) Soft-Margin Classifier.

**Loss Function of SVM:** loss function of SVM is given by

$$\text{Loss/Margin, } d = \operatorname{argmin}_{(p,b)} \|p\|/2 + C * \sum z_i$$

where  $d$  = margin/shortest distance between two hyperplanes.

The first term in the loss function measures the Margin-error and is known as Regularization-term.

$\sum z_i$  = Hinge loss or classification error

**Example:**

Consider the dataset with two classes Green and orange in two-dimensional space. **Our goal** is to classify any new datapoint into two classes based on the features of the datapoint.

- **Represent the data-points in 2d space (x,y):**  
e.g. Let suppose **Green points** as: (-1,0), (0,1), (0,-1), (1,0).  
and **Orange points** as: (3,0), (4,1), (4,-1).
- **Random select any hyperplane in 2D a line, that classifies these two classes.**  
e.g.  $w' * x + b = 2$
- **Locate/ select the support vectors and calculate the margin:**

From green class: (1,0) will be selected as Support Vector

e.g. Negative hyperplane  $w' * x + b = 1$

From orange class: (3,0) will be selected as Support Vector

e.g. Positive Hyperplane  $w' * x + b = 3$

Select the Hyperplane with Maximum Margin or Maximum shortest distance between the positive and negative hyperplane.

- Classification: for any new point  $x$ , if  $w'x+b<1$ ,  $x$  will be classified as negative or Green point and if  $w'x+b>3$ ,  $x$  will be classified as positive or Orange point.