# **Python Random shuffle() Method :**

### **Example :**

Shuffle a list (reorganize the order of the list items):

import random

mylist = ["apple", "banana", "cherry"]

random.shuffle(mylist)

print(mylist)

**Output :**

**['cherry', 'banana', 'apple']**

## **Definition and Usage**

The shuffle() method takes a sequence, like a list, and reorganize the order of the items.

## **Syntax :**

random.shuffle(*sequence*, *function*)

## **Parameter Values**

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| *sequence* | Required. A sequence. |
| *function* | Optional. The name of a function that returns a number between 0.0 and 1.0.  If not specified, the function [random()](https://www.w3schools.com/python/ref_random_random.asp) will be used |

**EXAMPLE:**

**import random**

**def myfunction():**

**return 0.1**

**mylist = ["apple", "banana", "cherry"]**

**random.shuffle(mylist, myfunction)**

**print(mylist)**

**Output:**

**["banana", "cherry", "apple"]**

**TASK – 6**

## **TASK 6 a) Write a Python program to shuffle the elements of a given list**

**AIM : Write a Python program to shuffle the elements of a given list**

**PROGRAM:**

**import random**

**nums = [1, 2, 3, 4, 5]**

**print("Original list:")**

**print(nums) random.shuffle(nums) print("Shuffle list:") print(nums)**

**words = ['red', 'black', 'green', 'blue']**

**print("\nOriginal list:")**

**print(words)**

**random.shuffle(words)**

**print("Shuffle list:")**

**print(words)**

**Sample Output:**

## **Python CSV Module**

**Original list:**

**[1, 2, 3, 4, 5]**

**Shuffle list:**

**[4, 2, 1, 5, 3]**

**Original list:**

**['red', 'black', 'green', 'blue']**

**Shuffle list:['green', 'blue', 'black', 'red']**

[Python](https://www.guru99.com/python-tutorials.html) provides a CSV module to handle CSV files. To read/write data, you need to loop through rows of the CSV. You need to use the split method to get data from specified columns.

## **CSV Module Functions**

In CSV module documentation you can find following functions:

* csv.field\_size\_limit – return maximum field size
* csv.get\_dialect – get the dialect which is associated with the name
* csv.list\_dialects – show all registered dialects
* csv.reader – read data from a csv file
* csv.register\_dialect – associate dialect with name
* csv.writer – write data to a csv file
* csv.unregister\_dialect – delete the dialect associated with the name the dialect registry
* **csv.QUOTE\_ALL** – Quote everything, regardless of type.
* **csv.QUOTE\_MINIMAL** – Quote fields with special characters
* **csv.QUOTE\_NONNUMERIC** – Quote all fields that aren’t numbers value
* **csv.QUOTE\_NONE** – Don’t quote anything in output

In this tutorial, we are going to focus only on the reader and writer functions which allow you to edit, modify, and manipulate the data in a CSV file.

## 

## **How to Read a CSV File in Python**

Below are steps to read CSV file in Python.

**Step 1)** To read data from CSV files, you must use the reader function to generate a reader object.

The reader function is developed to take each row of the file and make a list of all columns. Then, you have to choose the column you want the variable data for.

It sounds a lot more intricate than it is. Let’s take a look at this Python code to read CSV file, and we will find out that working with csv file isn’t so hard.

#import necessary modules

import csv

with open('X:\data.csv','rt')as f:

data = csv.reader(f)

for row in data:

print(row)

**Step 2)** When you execute the program above, the output will be:

['Programming language; Designed by; Appeared; Extension']

['Python; Guido van Rossum; 1991; .py']

['Java; James Gosling; 1995; .java']

['C++; Bjarne Stroustrup;1983;.cpp']

## **TASK 6 b).Write a Python program to read and display the content of a given CSV file?**

## **AIM: develop a Python program to read and display the content of a given CSV file**

**PROGRAM:**

import pandas as pd

grad\_url = "E:\Python Programs\Mall\_Customers.csv" grad\_students = pd.read\_csv(grad\_url)

#grad\_students

#Write your code here or in other code cells down print(grad\_students.shape) print(grad\_students.head())

## **OUTPUT:**