CherryPy

tutorialspoint

SIMPLY EASY LEARNING

www.tutorialspoint.com

https://www.facebook.com/tutorialspointindia

https://twitter.com/tutorialspoint

About the Tutorial

CherryPy allows developers to build web applications in much the same way they would build any other object-oriented Python program. This results in smaller source code developed in less time. It is being used in many production websites.

Audience

This tutorial is primarily designed for developers who work on Python and are required to create portable database-driven web-based applications.

CherryPy provides the CRUD (Create, Retrieve, Update and Delete) functionalities for applications and helps in managing the project from anywhere using the user's browser.

Prerequisites

To draw benefits from this tutorial, you need to have a basic knowledge of Python programming. A good understanding of Model-View-Controller and Object-Oriented Programming is also equally important. If you are not well aware of these concepts, we suggest you go through our short tutorial on Python.

Copyright & Disclaimer

All the content and graphics published in this e-book are the property of Tutorials Point (I) Pvt. Ltd. The user of this e-book is prohibited to reuse, retain, copy, distribute or republish any contents or a part of contents of this e-book in any manner without written consent of the publisher.

We strive to update the contents of our website and tutorials as timely and as precisely as possible, however, the contents may contain inaccuracies or errors. Tutorials Point (I) Pvt. Ltd. provides no guarantee regarding the accuracy, timeliness or completeness of our website or its contents including this tutorial. If you discover any errors on our website or in this tutorial, please notify us at contact@tutorialspoint.com

Table of Contents

1. CHERRYPY – INTRODUCTION

CherryPy is a web framework of Python which provides a friendly interface to the HTTP protocol for Python developers. It is also called a web application library.

CherryPy uses Python's strengths as a dynamic language to model and bind HTTP protocol into an API. It is one of the oldest web frameworks for Python, which provides clean interface and reliable platform.

History of CherryPy

Remi Delon released the first version of CherryPy in late June 2002. This was the starting point of a successful Python web library. Remi is a French hacker who has trusted Python for being one of the greatest alternatives for web application development.

The project developed by Remi attracted a number of developers who were interested in the approach. The approach included the following features:

- CherryPy was close to the model-view-controller pattern.
- A CherryPy class has to be processed and compiled by the CherryPy engine to produce a self-contained Python module embedding the complete application and also its own built-in web server.
- CherryPy can map a URL and its query string into a Python method call, for example:

```
http://somehost.net/echo?message=hello would map to echo(message='hello')
```

During the two years of development in CherryPy project, it was supported by the community and Remi released several improved versions.

In June 2004, a discussion started about the future of the project and whether it should continue with the same architecture. Brainstorming and discussion by several project regulars then led to the concept of object-publishing engine and filters, which soon became a core part of CherryPy2.Later, in October 2004, the first version of CherryPy 2 alpha was released as a proof of concept of these core ideas. CherryPy 2.0 was a real success; however, it was recognized that its design could still be improved, and needed refactoring.

After discussions based on feedbacks, CherryPy's API was further modified to improve its elegance, leading to the release of CherryPy 2.1.0 in October 2005. After various changes, the team released CherryPy 2.2.0 in April 2006.

Strengths of CherryPy

The following features of CherryPy are considered as its strengths:

Simplicity

Developing a project in CherryPy is a simple task with few lines of code developed as per the conventions and indentations of Python.

CherryPy is also very modular. The primary components are well managed with correct logic concept and parent classes are expandable to child classes.

Power

CherryPy leverages all the power of Python. It also provides tools and plugins, which are powerful extension points needed to develop world-class applications.

Open-source

CherryPy is an open-source Python Web Framework (licensed under the open-source BSD license), which means this framework can be used commercially at ZERO cost.

Community Help

It has a devoted community which provides complete support with various types of questions and answers. The community tries to give complete assistance to the developers starting from the beginner level to the advanced level.

Deployment

There are cost effective ways to deploy the application. CherryPy includes its own production-ready HTTP server to host your application. CherryPy can also be deployed on any WSGI-compliant gateway.

2. CHERRYPY – ENVIRONMENT SETUP

CherryPy comes in packages like most open-source projects, which can be downloaded and installed in various ways which are mentioned as follows:

- Using a Tarball
- Using easy_install
- Using Subversion

Requirements

The basic requirements for installation of CherryPy framework include:

- Python with version 2.4 or above
- CherryPy 3.0

Installing a Python module is considered an easy process. The installation includes the use of the following commands.

```
python setup.py build
python setup.py install
```

The packages of Python are stored in the following default directories:

- On UNIX or Linux,

  ```
  /usr/local/lib/python2.4/site-packages
  ```

  or

  ```
  /usr/lib/python2.4/site-packages
  ```

- On Microsoft Windows,

  ```
  C:\Python or C:\Python2x
  ```

- On Mac OS,

```
Python:Lib:site-package
```

Installation using Tarball

A Tarball is a compressed archive of files or a directory. The CherryPy framework provides a Tarball for each of its releases (alpha, beta, and stable).

It contains complete source code of the library. The name comes from the utility used in UNIX and other operating systems.

Here are the steps to be followed for the installation of CherryPy using tar ball:

Step 1: Download the version as per user requirements from http://download.cherrypy.org/.

Step 2: Search for the directory where Tarball has been downloaded and uncompress it. For Linux operating system, type the following command:

```
tar zxvf cherrypy-x.y.z.tgz
```

For Microsoft Windows, the user can use a utility such as 7-Zip or Winzip to uncompress the archive via a graphical interface.

Step 3: Move to the newly created directory and use the following command to build CherryPy:

```
python setup.py build
```

For the global installation, the following command should be used:

```
python setup.py install
```

Installation using easy_install

Python Enterprise Application Kit (PEAK) provides a python module named Easy Install. This facilitates deployment of the Python packages. This module simplifies the procedure of downloading, building and deploying Python application and products.

Easy Install needs to be installed in the system before installing CherryPy.

Step 1: Download the ez_setup.py module from http://peak.telecommunity.com/dist/ez_setup.py and run it using the administrative rights on the computer: python ez_setup.py.

Step 2: The following command is used to install Easy Install.

```
easy_install product_name
```

Step 3: easy_install will search the Python Package Index (PyPI) to find the given product. PyPI is a centralized repository of information for all Python products.

Use the following command to deploy the latest available version of CherryPy:

```
easy_install cherrypy
```

Step 4: easy_install will then download CherryPy, build, and install it globally to your Python environment.

Installation using Subversion

Installation of CherryPy using Subversion is recommended in the following situations:

- A feature exists or a bug has been fixed and is only available in code under development.
- When the developer works on CherryPy itself.
- When the user needs a branch from the main branch in the versioning control repository.
- For bug fixing of the previous release.

The basic principle of subversioning is to register a repository and keep a track of each of the versions, which include a series of changes in them.

Follow these steps to understand the installation of CherryPy using Subversion:

Step 1: To use the most recent version of the project, it is necessary to check out the trunk folder found on the Subversion repository.

Step 2: Enter the following command from a shell:

```
svn co http://svn.cherrypy.org/trunk cherrypy
```

Step 3: Now, create a CherryPy directory and download the complete source code into it.