

Training: Python Academy
Twisted

TRAINING TERMS

2025-09-10 | 3 days | Kraków / Virtual Classroom

2025-10-08 | 3 days | Warszawa / Virtual Classroom

TRAINING GOALS:

This tutorial is intended for programmers with at least some **Python** experience who want to use the **Twisted framework**. It is suitable for people trying to start new Twisted projects, are looking to integrate Twisted with an existing non-Twisted project, or who have inherited an existing Twisted project.

CONSPECT:

- Core course
 - Introduction
 - Event loops
 - Environment setup
 - Dependencies and required software
 - Where to get help
 - Documentation
 - Support channels: IRC, mailing list...
 - Protocols, factories and endpoints
 - Introduction
 - Building a simple protocol
 - Client and server factories
 - Serving your protocol:
 - reactor.listen/connect
 - Endpoints
 - Existing and composable protocols
 - Deferreds
 - Introduction
 - Exercises

- @inlineCallbacks
- Important toplevel packages
 - Twisted Web
 - Servers (with exercises)
 - Resource traversal
 - Clients (exercises based on previous servers)
 - Twisted Cred: pluggable authentication
 - Overview
 - Exercises
- Other toplevel packages and third party packages
 - twisted.words: Chat protocols & IRC support
 - IRC clients
 - A simple bot
 - Adding more advanced behavior (e.g. webhooks)
 - Building an IRC server
 - Wokkel (Friend fork with advanced Jabber/XMPP support)
 - twisted.news: NTTP support
 - twisted.conch: SSH support (timing depends on level of detail and which components)
 - Database access
 - twisted.enterprise + txpostgres
 - txredis, paisley (couchdb), telephus (cassandra)
 - Axiom
 - twisted.spread: distributable computing
 - twisted.names: DNS servers and clients
 - Twisted Community Code (tx) (length depends on the package)
 - scrapy: advanced scraping architecture built on Twisted
- Advanced topics
 - Dealing with legacy/synchronous code
 - WSGI app case study: django-packages
 - Threads: deferToThread(Pool), blockingCallFromThread
 - Dealing with CPU-bound problems
 - Introduction
 - Explicit cooperative multitasking
 - Threads (reference; see previous mention of threads)
 - Multiprocessing options and Ampoule
 - Application framework

- Services and applications
- twisted: running servers & daemonization
- Twisted's plugin system
- Writing your own twisted plugins
- Advanced application concepts
- Process monitoring: daemontools, upstart, systemd, etc.
- Live server monitoring and maintenance with manhole
- Contributing to Twisted
- Issue tracker
- Setting up the environment
- Review process

In addition to the previous topics, the following topics are touched on as needed:

- Test Driven Twisted: twisted.trial (tutorial is TDD where sensible)
- twisted.python: useful random python utilities
- FilePath: like paths but better
- Logging

REQUIREMENTS:

- Basic knowledge of **Python** and **object oriented programming** is required.
- If you don't know Python it is strongly recommended to take [Python for Programmers](#) before this course.

Difficulty level



CERTIFICATE:

The participants will obtain certificates signed by Python Academy.

TRAINER:

Authorized Python Academy Trainer.