To discuss this course and customizations:
Call: 434-509-5680 or Email: sales@cloudcontraptions.com

Distributed Task Automation with Python Celery and RabbitMQ

Duration

16 hours

Target Audience

- All students should have taken the Python Task Automation course or have significant experience with the topics covered in the Python Task Automation course.
- Experience with RabbitMQ and PostgreSQL is helpful but not required.

Executive Summary

This course gives Python developers hands-on mastery in distributed task automation using Celery and RabbitMQ. Learn to set up scalable environments with Docker, automate tasks with Celery, and manage messaging workflows with RabbitMQ and PostgreSQL. Build, monitor, and deploy robust task queues, tackle advanced topics like routing and scheduling, and scale your solutions for production—ideal for anyone ready to automate complex workflows and boost backend efficiency.

Description

This comprehensive course is tailored for programming professionals eager to explore Distributed Task Automation, offering an in-depth journey through Celery and RabbitMQ. Beginning with environment setup in Visual Studio Code—leveraging Python and Docker—it covers containerization principles and remote data storage, guiding you through running PostgreSQL and RabbitMQ in containers. You'll gain hands-on experience with Celery from foundational concepts to advanced techniques, and conclude with practical strategies for scaling and deploying Celery in production. Whether you're a seasoned developer or new to programming, this customizable course equips you with the skills and knowledge to automate distributed tasks effectively, with real-world examples relevant to your domain.

To discuss this course and customizations:
Call: 434-509-5680 or Email: sales@cloudcontraptions.com

Objectives

- Understand the concept of Distributed Task Automation and learn about tools like Celery and RabbitMQ.
- Set up a development environment with Visual Studio Code, Python, and Docker.
- Learn about containerization using Docker and Docker Compose.
- Understand remote data storage and how to run PostgreSQL in a container.
- Learn about Message Brokers and how to run RabbitMQ in a container.
- Gain a solid understanding of Celery basics, including task definition, running tasks, and worker management.
- Delve into advanced Celery concepts like task serialization, routing, and scheduling periodic tasks.
- Learn how to scale and deploy Celery in a production environment, including load balancing, high availability, and performance tuning.

Duration

16 hours of intensive training with live instruction delivered over two to four days to accommodate varied scheduling needs.

Training Materials

All students receive comprehensive courseware covering all topics in the course. The instructor distributes courseware via GitHub. The courseware includes documentation and extensive code samples. Students practice the topics covered through challenging hands-on lab exercises. Students will need a free, personal GitHub account to access the courseware. All students will need a modern web browser such as Google Chrome. Student machines will need a text editor like Visual Studio Code, the latest Python version, Docker Desktop, PanDoc, and OpenOffice. Students will need permission to install NPM and PyPi packages as well as the ability to download Docker images. Preconfigured student virtual machines can provided upon request.

Software Requirements

Students will need a free, personal GitHub account to access the courseware. Student machines will need a text editor like Visual Studio Code, the latest Python version, Docker Desktop, PanDoc, and OpenOffice. Students will need permission to install NPM and PyPi packages as well as the ability to

download Docker images. Preconfigured student virtual machines can provided upon request.

Training Topics

Overview of Distributed Task Automation
- What is Distributed Task Automation?
- Overview of Celery and its Features
- Celery compared to Faust
- What is Message Broker?
- What is RabbitMQ?
- RabbitMQ + PostgreSQL compared to Kafka + Zookeeper

Development Environment
- Configure Visual Studio Code for Python Script Programming
- Python Code Linting & Reformatting with Ruff & MyPy
- Debugging Python Scripts with Visual Studio Code
- Docker Desktop

Containerization
- What is a Container?
- What is Docker?
- What is Docker Hub?
- Images and Containers
- Create an Image with Dockerfile
- Run Containers
- Configure Containers with Environment Variables
- Docker Compose
- Docker Compose Networking
- Docker Compose Volume

Remote Data Storage
- What is Remote Data Storage?
- Running a Data Storage in a Container
- Running PostgreSQL in a Container
- Running PostgreSQL Client Tool in a Container
- Configure with Docker Compose
- Read/Write Data to PostgreSQL with Python SQLAlchemy

Remote Message Broker
- What is a Message Broker?
- Running a Message Broker in a Container
- Running RabbitMQ in a Container
- Running RabbitMQ Client Tool in a Container
- Configure with Docker Compose
- Read/Write Data to RabbitMQ with Python and Pika

Building a REST API with Flask
- Create an Application
- Define a Hello World Route
- Configure a Path Parameter
- Configure a Query String Parameter
- Reading Request Body
- Writing a Response Body
- Handling Cookies
- Handle Errors
- Return Specific Status Codes
- Asynchronous Routes
- Serving Static Files

Database Access
- Connecting to a Database
- Querying Data
- Modifying Data
- Transactions
- SQLAlchemy
- Running a Database with Docker

Document and Version APIs
- Swagger
- OpenAPI
- Versioning

Security
- Secure a REST API with JWT
- Configure CORS

Cloud Contraptions
TRAINING & CONSULTING

To discuss this course and customizations:
Call: 434-509-5680 or Email: sales@cloudcontraptions.com

Cloud Deployment
- Microservice Architecture
- REST API Containerization
- Running Server-Less in Azure