

Implementing Knowledge Graph Technology

for Question Answering System

FA-0494

Software Development · DevOps, Cloud & Infrastructure · Data & Analytics

Introduction

This 1 + 3 + 1 day course provides an introduction to semantic technology and its application in building a Q & A app with Python. Participants will learn about natural language processing, knowledge graphs, and the Semantic Web. They will also gain hands-on experience in using Python libraries for semantic technology, creating a knowledge graph, and integrating it with a FastAPI app to create a Q & A app.

Learning Outcome

By the end of the course, participants will be able to:

- Understand the basics of semantic technology and its role in Q & A applications
- Use Python libraries for natural language processing and semantic technology
- Create a knowledge graph using RDF and OWL
- Build SPARQL queries to retrieve information from a knowledge graph
- Implement a Flask app to serve as the front-end of a Q & A app
- Integrate the knowledge graph and natural language processing into a Q & A app
- Deploy a Q & A app using a cloud service provider
- Evaluate and improve a Q & A app based on semantic technology

Duration : 4 Days

Prerequisites

- Basic programming knowledge, including familiarity with Python
- Understanding of web development concepts, such as HTML and CSS
- Basic understanding of database concepts, such as SQL
- Familiarity with Linux command line interface and basic shell scripting
- Completion of the Course "Knowledge Graph Technology for Question Answering System."

Additionally, some prior exposure to natural language processing and web ontology languages such as RDF and OWL would be helpful, but not required.

Training outline

1. Day 1:

- Prerequisites overview
- Python recap
- Python app deployment Recap
- Project Overview

2. Day 2:

- Installing and setting up prerequisites and required libraries
- Exploring natural language processing using Python
- Creating a simple Q & A app without semantic technology

3. Day 3:

- Building a simple knowledge graph using RDF and OWL
- Creating SPARQL queries to retrieve information from the knowledge graph
- Integrating the knowledge graph into the Q & A app

4. Day 4:

- Building a FastAPI app to serve as the front-end of the Q & A app
- Integrating the knowledge graph and natural language processing into the app

- Deploying the Q & A app using a cloud service provider

Conclusion:

- Recap of the concepts covered in the course
- Discussion of possible extensions or improvements to the Q & A app
- Recommendations for further learning in semantic technology and Python

5. Day 5:

Assessment

1. Create a Simple Knowledge Graph: Provide students with a dataset and ask them to create a simple knowledge graph using RDF and SPARQL. The knowledge graph should include a hierarchy of concepts, relationships between entities, and properties for each entity.
2. Querying a Knowledge Graph: Provide students with the knowledge graph from exercise 1 and ask them to write SPARQL queries to retrieve specific information from the graph. The queries should demonstrate understanding of the RDF data model and basic SPARQL syntax.
3. Using Knowledge Graphs for Search and Recommendation: Ask students to research a company or organization that uses knowledge graphs for search or recommendation and write a short report on how they use the technology. The report should demonstrate an understanding of the benefits and challenges of using knowledge graphs for these applications.
4. Knowledge Graphs in Action: Provide students with a real-world use case of a knowledge graph and ask them to analyze it using SPARQL queries. The use case should demonstrate how knowledge graphs can be used to solve real-world problems and how to query and analyze the data in the graph.