

Web Services

Assignment 1

Dr. Kyle Goslin

Weight: 25%

Due date: Check course page under Assignment 1

Introduction

A company has contacted you and asked that you create a complete API solution for inventory management. They have provided a dump of their inventory system in **CSV format**.

Each product in their inventory has the following attributes:

ProductID	Name	UnitPrice	StockQuantity	Description
-----------	------	-----------	---------------	-------------

This sample **auto_products.csv** file can be downloaded from the course page under the heading **Assignment 1**.

You are tasked with building a **complete API** that utilizes a MongoDB database to store the data, **FastAPI** to create the API. To aid your development with DevOps, use **Jenkins** to speed up the delivery of your application to the customer.

System Specifics

This section outlines each core feature that should be added to your project. Use this as a checklist for the project to ensure you have each component completed.

Before you start, remember to use a Python Virtual Environment (venv) to make your life easier!

	Data setup
1	The CSV data should be converted into a JSON format , and stored in either a local or a cloud-based MongoDB database.
	API Development

2	<p>The API should have the following endpoints:</p> <ul style="list-style-type: none"> • /getSingleProduct - This allows you to pass a single ID number into the endpoint and return the details of the single product in JSON format. • /getAll - This endpoint should return all inventory in JSON format. • /addNew - This endpoint should take in all 5 attributes of a new item and insert them into the database. • /deleteOne - This endpoint should delete a product by the provided ID. • /startsWith - This should allow the user to pass a letter to the URL, such as s, and return all products that start with s. • /paginate - This URL should pass in a product ID to start from and a product ID to end from. The products should be returned in a batch of 10. • /convert - All of the prices are currently in dollars in the sample data. Implement a URL titled /convert which takes in the ID number of a product and returns the price in euro. An online API should be used to get the current exchange rate.
	For all variables that are passed in to your URLs, ensure data type checks are performed using Pydantic .
	Jenkins
3	Implement Python unit tests for each of the URLs created to show that they are working correctly when the tests are run.
	Generate a PDF file that contains the results from your unit tests automatically.
	Commit your final code to a GitHub repository.
	Set up a complete Jenkins workflow to pull the code from GitHub, run the Python unit tests on the API that you have created.

	Create a zip that contains a database dump of product records from your MongoDB database automatically. The filename of the zip should be database-DATE-TIME.zip where the current date and time are included.
	A README.txt file should be created automatically that contains a list of all your API endpoints showing the parameters that can be passed to it. A reference to the FastAPI document should also be included.
	Create a PDF output of your unit tests automatically. This PDF should be added to the final zip file created.
	At the end of your Jenkins workflow, it should create a zip file containing the API source code zip file and the database dump zip. The filename of the zip should be complete-DATE-TIME.zip where the current date and time are included.

Tip: Use **Postman** to test your API endpoints. A web browser only allows you to run GET requests, where with Postman we can send POST requests and manually specify what attributes and values to send.

Deliverable

Create a PDF document containing:

1. Your Fast API Python code.
2. Your unit test code.
3. Screenshots of all the URL endpoints for your API being run.
4. Screenshots of your Jenkins pipeline.
5. Screenshots showing the contents of your final zip file created by Jenkins.

You are required for this assignment to demo your final implementation and answer questions about all the technologies used, and the reasoning behind your choices.