How Python 3.9 fixed decorators and improved dictionaries

This is the tenth in a series of articles about features that first appeared in a version of Python 3.x. Some of these versions have been out for a while. Python 3.9 was first released in 2020 with cool new features that are still underused. Here are three of them.

Adding dictionaries

Say you have a dictionary with "defaults," and you want to update it with parameters. Before Python 3.9, the best option was to copy the defaults dictionary and then use the .update() method.

Best Python Frameworks To Create Mobile Apps and Games

This article will cover a list of useful Python frameworks that can be used to develop apps and games for mobile devices. Some of these frameworks also support desktop apps or work as standalone build tools for compiling builds for both desktop and mobile devices. You can use the same code base with minor modifications to deploy apps and games to both desktop and mobile devices.

Introduction to machine learning with Jupyter notebooks | Red Hat Developer

Recently, I was working on an edge computing demo that uses machine learning (ML) to detect anomalies at a manufacturing site. This demo is part of the AI/ML Industrial Edge Solution Blueprint announced last year. As stated in the documentation on GitHub, the
blueprint enables declarative specifications that can be organized in layers and that define all the components used within an edge reference architecture, such as hardware, software, management tools, and tooling.

At the beginning of the project, I had only a general understanding of machine learning and lacked the practitioner's knowledge to do something useful with it. Similarly, I'd heard of Jupyter notebooks but didn't really know what they were or how to use one.

This article is geared toward developers who want to understand machine learning and how to carry it out with a Jupyter notebook. You'll learn about Jupyter notebooks by building a machine learning model to detect anomalies in the vibration data for pumps used in a factory. An example notebook will be used to explain the notebook concepts and workflow. There are plenty of great resources available if you want to learn how to build ML models.

- **Pyodide: Python for the browser [LWN.net]** [5]

  Python in the browser has long been an item on the wish list of many in the Python community. At this point, though, JavaScript has well-cemented its role as the language embedded into the web and its browsers. The Pyodide project provides a way to run Python in the browser by compiling the existing CPython interpreter to WebAssembly and running that binary within the browser's JavaScript environment. Pyodide came about as part of Mozilla's Iodide project, which has fallen by the wayside, but Pyodide is now being spun out as a community-driven project.

- **How to Implement a WebSocket in Python ? Linux Hint** [6]

  WebSocket is an application layer protocol that allows two-way communication between a client and a server. WebSocket protocol works first by creating a handshake and then a message framing implemented over TCP rather than basic HTTP.

  WebSockets and other similar technologies such as SSE (Server-Sent Events) and WebRTC are helpful in applications where the server needs to keep an open connection to the connected clients. An excellent example of WebSockets used in applications is a chat application, online multiplayer games, and real-time tools such as analytics and collaboration tools.

  WebSockets provides us with a full-duplex, bidirectional connection between the server and the connected clients over the Web. That means both the server and the client can push data once there is an established connection.

  In this tutorial, I will not dive deep into how WebSockets work. Instead, I will show you how you can use Python to implement a simple application using WebSocket.

  If you wish to learn more about HTTP, WebSocket, and SSEs, check out the other tutorials on
NOTE: Before we begin, we assume you are familiar with basic networking concepts such as HTTP and HTTP requests. To implement the concepts in this tutorial with a degree of ease, you need to have basic Python and JavaScript Programming knowledge.