Hello and welcome to this Icon Data API tutorial. My name is Eve, I'm the founder and managing partner of The Python Company. Today's tutorial is about financial data science, and in particular, we will cover portfolio theory, expected returns, and expected volatility. We will use the Icon Data API to retrieve data and Python, specifically Pandas, to derive statistics and simulate portfolio compositions in detail.

The agenda for today's topic includes the following steps:

Introduction to Portfolio Selection: We will discuss the cornerstone of quantitative finance, Markowitz's portfolio theory, which formalizes the investment process and explains market behavior.

Reading Historical Cross-Asset Data: Using the Icon Data API, we will retrieve historical close prices for a set of financial instruments, including Apple stock, Amazon stock, SP500 ETF, Gold ETF, and Euro/US Dollar exchange rate.

Calculating Statistics for Single Instruments: We will compute the log returns for each financial instrument, which will be the basis for other statistics.

Portfolio Statistics: We will calculate portfolio statistics, such as expected returns and expected volatility, based on the composition of different risky assets.

Portfolio Composition and Expected Volatility: Using a Monte Carlo simulation, we will randomly simulate portfolio compositions and observe their expected returns and volatility.

Deriving the Minimum Volatility Portfolio: We will implement a minimization procedure to find the portfolio with the lowest volatility within certain constraints.

Let's get started with the Python code. We will begin by importing the necessary packages, including the Icon Python wrapper to connect to the Icon Data API. The data API must be running in the background, either as a proxy or the desktop application itself. We will define a small universe of financial instruments comprising five different ones: Apple stock, Amazon stock, SP500 ETF, Gold ETF, and Euro/US Dollar exchange rate.

Next, we will retrieve the historical close prices for these financial instruments for the period from the beginning of 2017 until mid-February 2018. Afterward, we will calculate the log returns and normalize the time series data for each instrument.

In mean-variance portfolio theory, the average returns of financial instruments play an important role as they are used to approximate the expected return. We will calculate the mean returns for each instrument, scaling them to an annualized basis for better comparison. Additionally, we will compute the historical volatilities, which represent the standard deviation of returns, also scaled to an annualized basis.

Moving on to portfolio statistics, we will start with a small portfolio containing just two financial instruments: Apple stock and Amazon stock. Even though these two are positively correlated as technology stocks, we will simulate portfolio compositions using random weights and observe their expected returns and volatility.

Expanding the portfolio, we will include all five financial instruments and perform multiple simulations to observe the distribution of expected returns and volatility. We will notice that diversification allows us to achieve relatively low volatility levels by including low-risk instruments in the portfolio.

Finally, we will derive the minimum volatility portfolio using a minimization procedure. This portfolio will have the lowest volatility among all possible compositions with certain constraints, such as not allowing short selling. The optimal portfolio composition will likely include low-risk instruments like SP500 ETF, Gold ETF, and Euro/US Dollar exchange rate.

In conclusion, we have covered various aspects of portfolio theory and how to use the Icon Data API and Python to analyze and simulate portfolio compositions. The combination of Pandas and Cufflinks enables powerful data visualization, while NumPy and SciPy provide robust optimization and statistical tools for efficient implementation. I hope this tutorial has been illuminating, and I wish you happy Python coding and successful financial data analysis. See you in the next tutorial! Bye bye!