HW 12 Design Patterns Summary

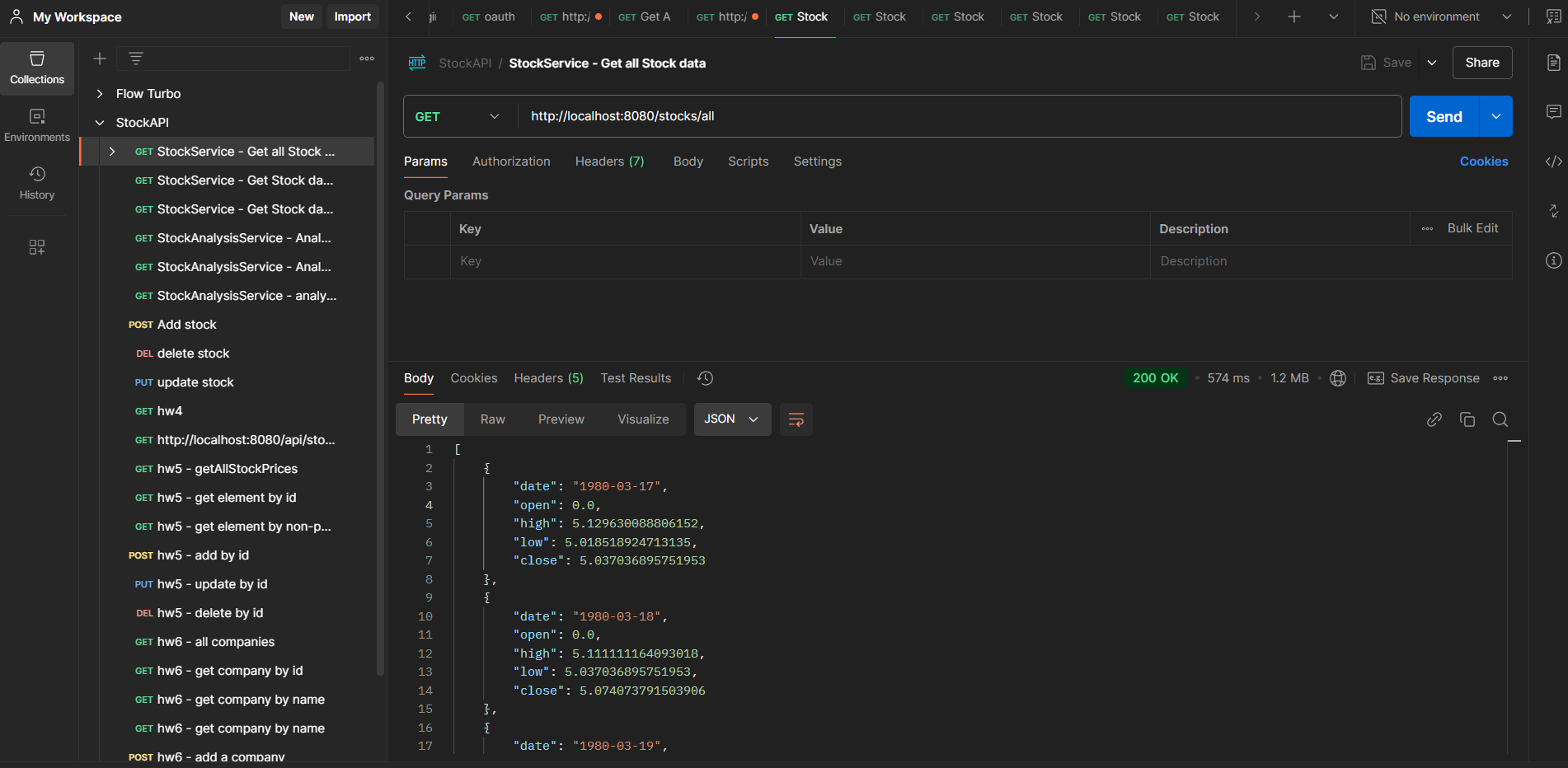
In this homework, I implemented two design patterns: the **Factory Pattern** and the **Strategy** **Pattern** on HW2\_REST\_API ([github link](https://github.com/deep3072/Homework_ACS560)). The Factory Pattern was used to encapsulate the creation logic of StockData instances. I created a StockDataFactory class that provides a method to create StockData objects from CSV data. This approach centralizes the instantiation logic, making the code more modular and easier to maintain. The StockRepository class acts as the client that uses the factory to obtain instances of StockData, ensuring that the creation logic is decoupled from the repository logic.

The Strategy Pattern was applied to encapsulate different stock analysis algorithms. I defined a StockAnalysisStrategy interface and implemented various strategies such as MeanStrategy, MedianStrategy, ModeStrategy, and AverageStrategy. The StockAnalysisServiceImpl class uses these strategies to perform the required calculations. This pattern allows for easy extension and modification of analysis algorithms without altering the core logic of the service class. By encapsulating each calculation in its own strategy class, this implementation adheres to the Single Responsibility Principle, making the code more flexible, maintainable, and testable.

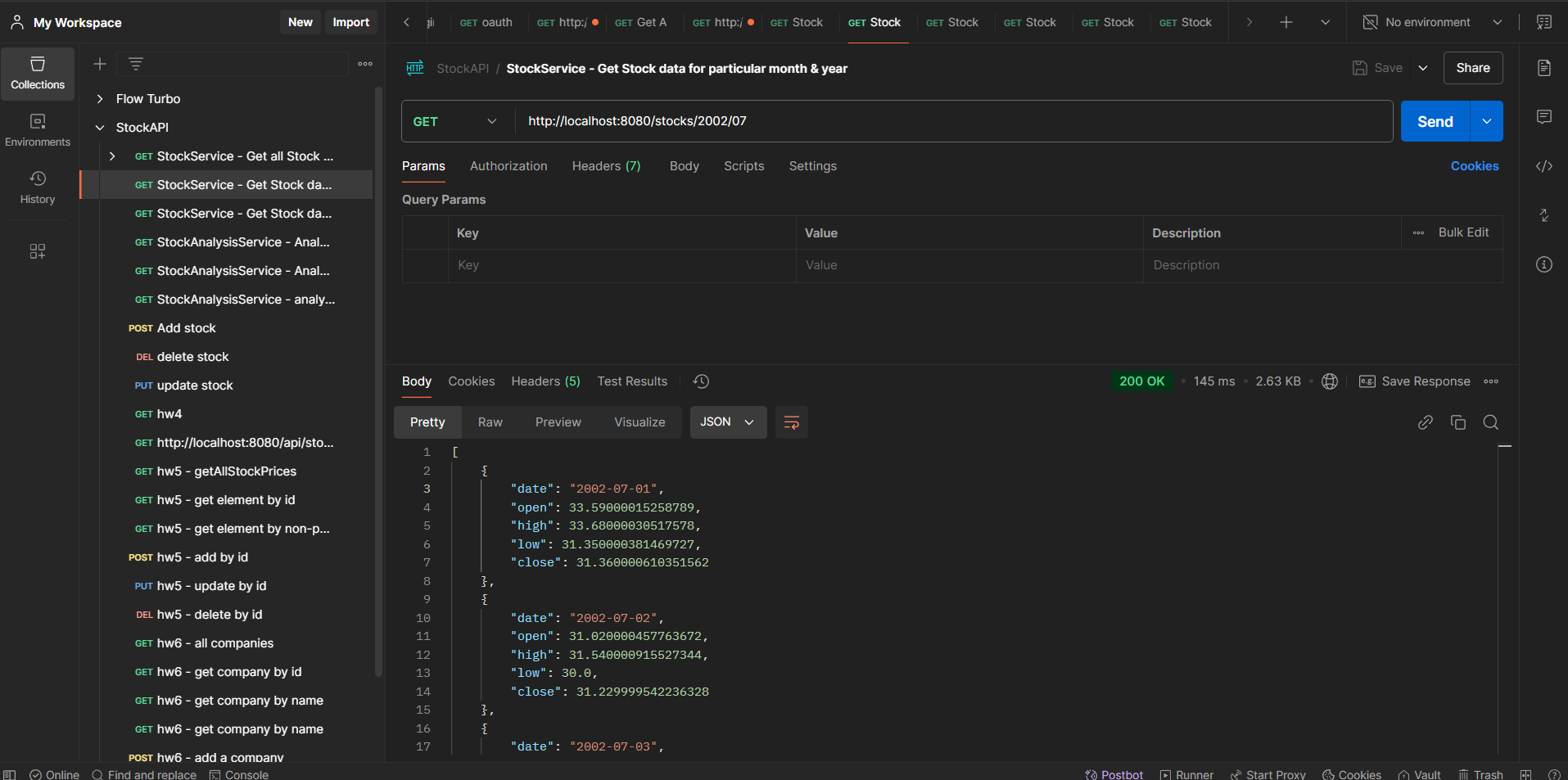
The impact of these implementations is significant. The Factory Pattern improves code readability and maintainability by centralizing object creation. It also makes it easier to manage changes to the instantiation logic. The Strategy Pattern enhances the flexibility and extensibility of the stock analysis functionality. New analysis methods can be added without modifying existing code, and different strategies can be used interchangeably. Overall, these design patterns contribute to a cleaner, more modular, and more maintainable codebase, facilitating future development and maintenance efforts.

**Screenshots:**

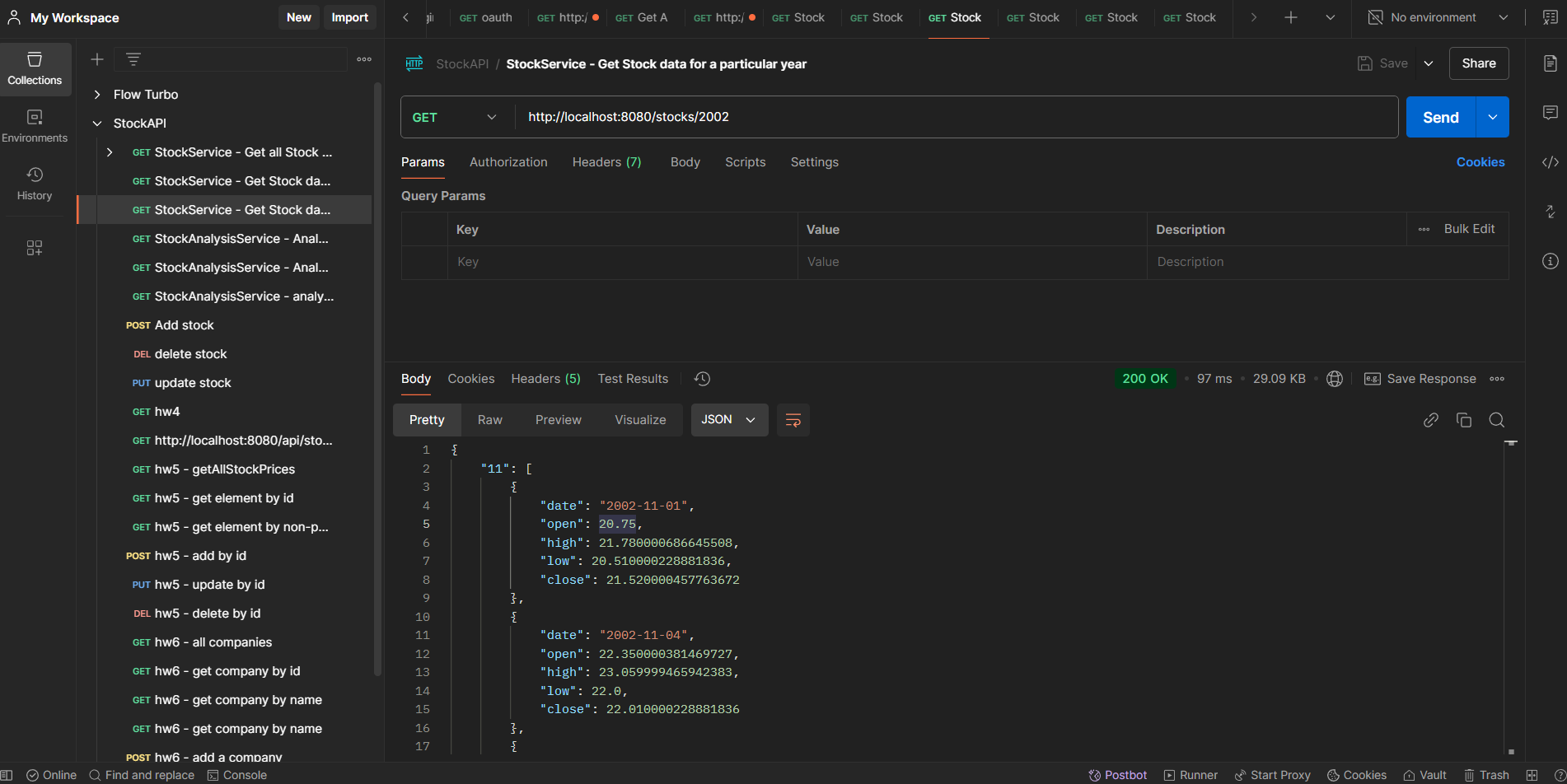
1. **StockService - Get all Stock data**

****

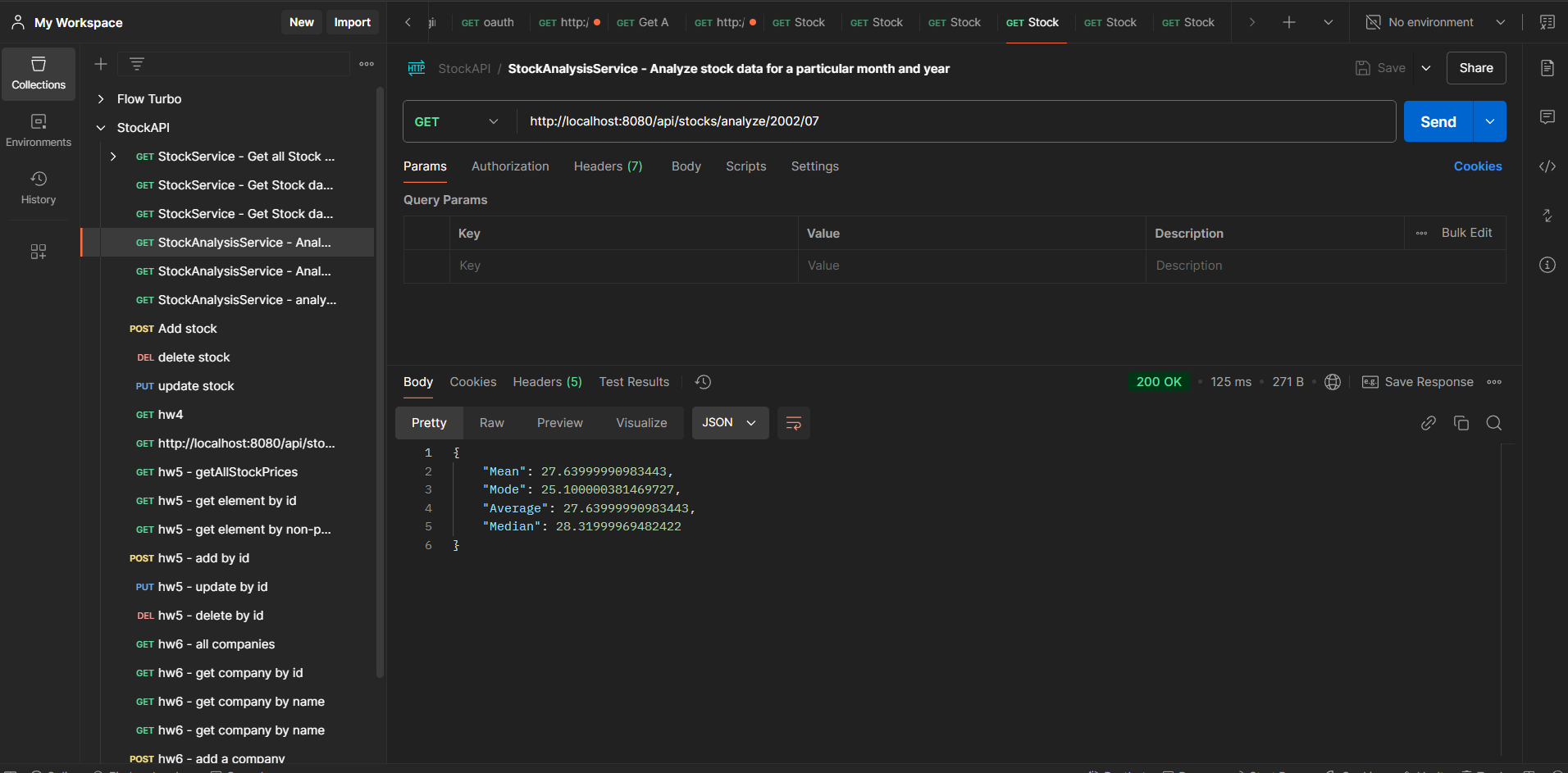
**2. StockService - Get Stock data for particular month & year**

****

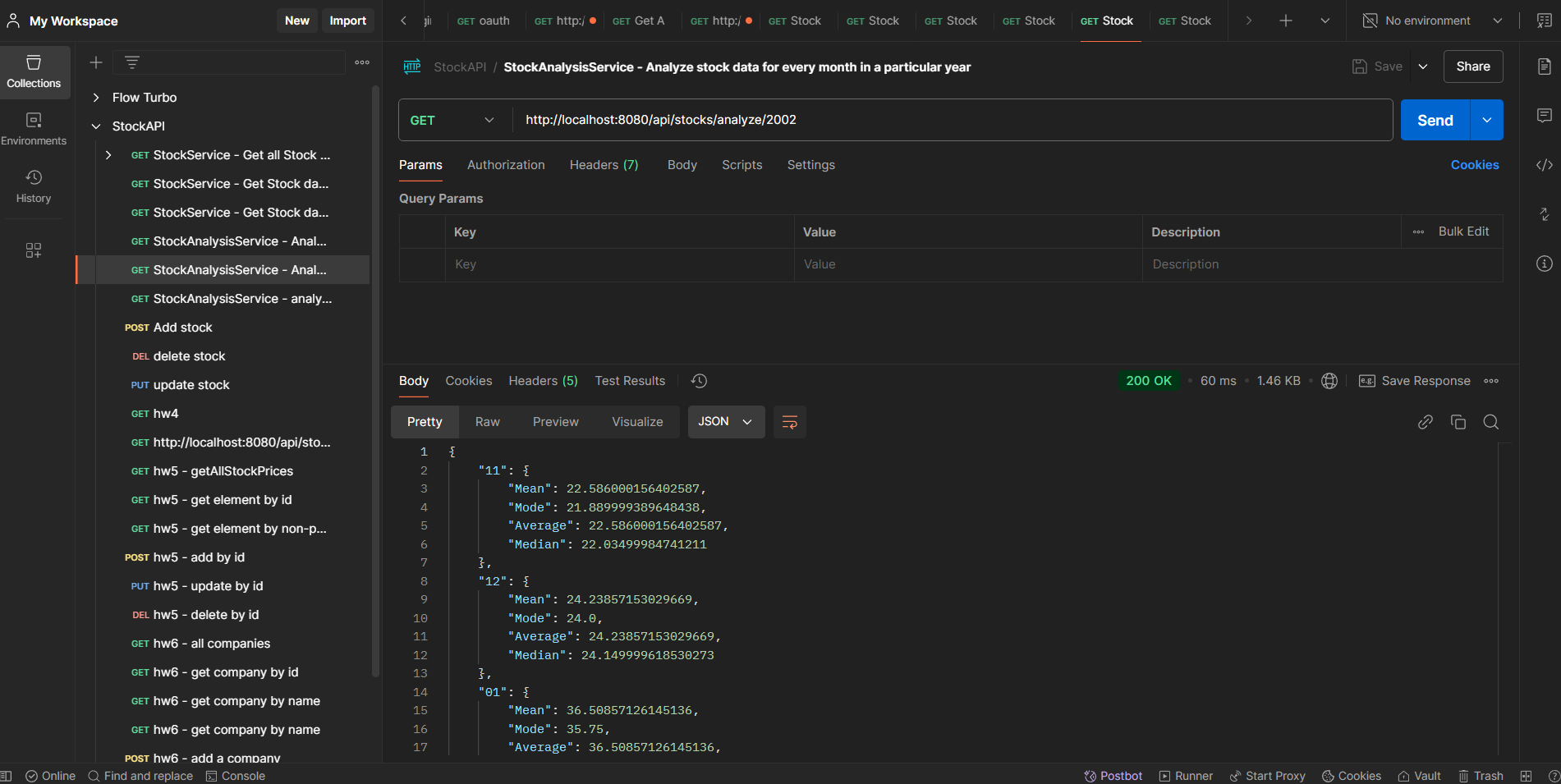
**3. StockService - Get Stock data for a particular year**

****

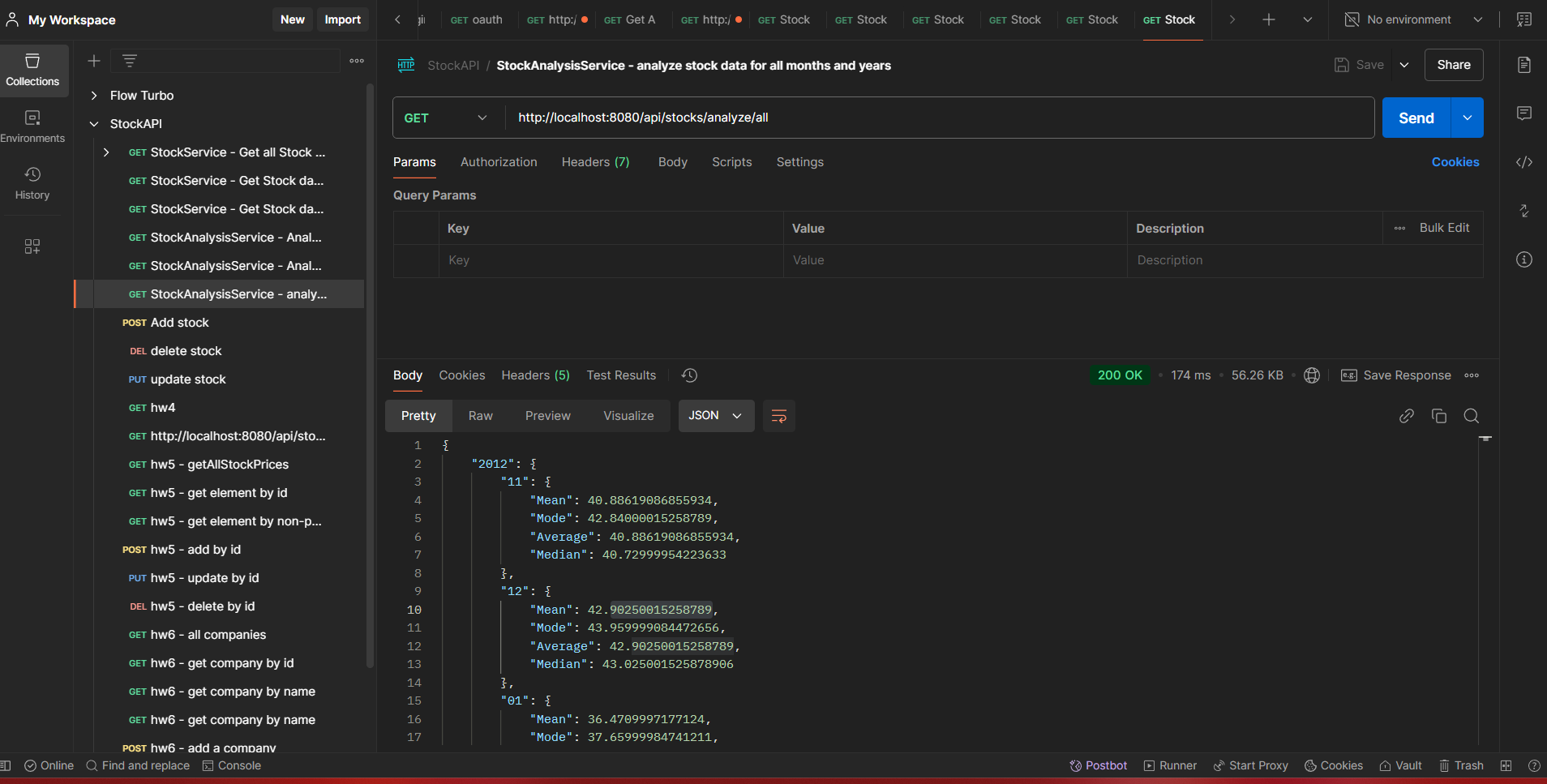
**4. StockAnalysisService - Analyze stock data for a particular month and year**

****

**5. StockAnalysisService - Analyze stock data for every month in a particular year**

****

**6. StockAnalysisService - analyze stock data for all months and years**

****