**Automated Trading** is the terminology given to trade entries and exits that are processed and executed via a computer. Automated trading has certain advantages:

1. **Minimizes human intervention:** Automated trading systems eliminate emotions during trading. Traders usually have an easier time sticking to the strategy by holding emotions in check.
2. **Backtesting on historical data:** Backtesting allows a trader to employ their strategies on historical data (previous week/ month/ year data). This helps them understand the implications of their strategy on real-time data and help them determine the probability of winning or losing trade.
3.  **Preserving Discipline in a volatile market:** When the market is volatile, traders jump the trading rules. This brings about indiscipline in the market. Discipline is lost when a trader gives in to the human emotion of greed or fear. Such parameters can be avoided with Automated trading. Automated trading helps ensure the maintenance of consistency, ensuring that execution of strategy follows rules.
4.  **Increases the speed of order entry:** Because computers respond to changing market conditions instantly, automated systems may produce orders as soon as the trade requirements are met.

[Python](https://www.geeksforgeeks.org/python-programming-language/) gained popularity in quant finance due to its capacity of building intricate statistical models with ease. This is because of scientific libraries like Pandas, NumPy, Matplotlib, PyAlgoTrade, Pybacktest, and more.

**Components Required for Automated Trading**

1. **Anaconda:** The first step in setting up Python is downloading Anaconda. Anaconda is a reliable distribution of Python, it consists of all tools and libraries, required to execute a Python code.
2. **Spyder IDE:** IDE stands for Integrated Development Environment. It provides an interface to write, debug, compile and execute the Python code.
3. **Jupyter Notebook:** Jupyter notebook is an interactive platform used to understand code snippets. Jupyter Notebook primarily uses ‘markdown’ cells for the explanation of the code and ‘code’ cells for the execution of the code. It’s highly useful for learners who are trying to understand the working of code snippets.

**Note:** Spyder IDE is used to run and execute big projects whereas Jupyter Notebook is used to execute small chunks of code.

### Understanding popular Packages/ Libraries in Python with respect to Quant Trading

Python has a large library collection that can be used for different functionalities such as programming, machine learning, visualization, etc. However, before actually getting started with Python, we’ll talk about the most important libraries needed for coding trading strategies.

We will need to import financial data, conduct the numerical assessment, construct trading strategies, plot graphs, and perform data backtesting. The Libraries needed are listed below:

 **NumPy:** NumPy shortened for NumericalPy is used for numerical analysis of data.

 **Pandas:** Pandas is widely used when working with data in tabular format (i.e rows and columns) such as spreadsheets. It can be used to import Excel and CSV files in the python code.

 **Matplotlib:** This library comprises functions used for plotting 2D graphs.

 **TA-Lib:** TA-Lib is extensively used to perform technical analysis on data such as Bollinger Bands, RSI (Relative Strength Indicator), VWAP (Volume Weighted Average), MA (Moving Average).

 **Zipline:** An event-driven system that supports backtesting and live trading.