In debugging code, there are three commonly used step commands: step into, step over, and step out. These commands control the flow of execution when stepping through the code line by line.

1. **Step Into** (`F11` in most IDEs):
   - When you step into a function or method call, the debugger takes you inside that function or method, allowing you to step through its code line by line.
   - This is useful when you want to inspect the internal behavior of a function or method in detail.

2. **Step Over** (`F10` in most IDEs):
   - When you step over a function or method call, the debugger executes the entire function or method without stepping into it.
   - The debugger moves to the next line of code after the function or method call, treating the called function or method as a single step.
   - This is useful when you are not interested in the internal implementation details of a particular function or method and want to quickly move past it.

3. **Step Out** (`Shift + F11` in most IDEs):
   - When you step out, the debugger continues executing the current function or method until it returns to the caller, at which point it stops at the next line of code after the function or method call.
   - This is useful when you have stepped into a function or method and want to quickly return to the caller without stepping through the remaining lines of code in the current function or method.

In summary:

- **Step Into**: Follows the execution into the called function or method.
- **Step Over**: Executes the called function or method without stepping into it.
- **Step Out**: Continues executing the current function or method until it returns to the caller.

These step commands allow you to control the granularity of your debugging process, enabling you to focus on specific areas of your code or skip over parts that you are not interested in debugging at the moment.