**@MyBaseListener**

This code is a Java class that extends JavaParserBaseListener from the ANTLR library. It overrides two methods enterTypeDeclaration and enterBlock, and also has a constructor that takes a TokenStreamRewriter object as input.

In the enterTypeDeclaration method, the code checks if add\_imports is false. If it is, it inserts two import statements for java.io.FileWriter and java.util.\* before the start of the TypeDeclarationContext. It then sets add\_imports to true.

In the enterBlock method, the code inserts a block of code after the start of each BlockContext. This code creates a new FileWriter object, writes a string containing the block number to a file named visited.txt, and closes the FileWriter object. The block number is incremented for each block visited.  
==========================

**/src/Main.java**

This is a Java program that uses ANTLR (ANother Tool for Language Recognition) to parse a Java source code file and insert code to log the visit of each block in the file.

The program starts by importing the necessary ANTLR libraries and other required libraries, such as the SortedSet and Scanner classes from the Java standard library. It then prompts the user to enter the file path of the Java source code file to be parsed.

The program **reads the input file** as an ANTLRInputStream, creates a lexer using the JavaLexer class, and passes the lexer output to a CommonTokenStream. This token stream is then passed to a JavaParser, which generates a parse tree by parsing the token stream.

Next, the program creates a TokenStreamRewriter object that will be used to modify the token stream. **It then creates a ParseTreeWalker and calls its walk() method**, passing in an instance of a custom listener class called **myBaseListener**. This listener class extends the JavaParserBaseListener and overrides the enterBlock() and enterTypeDeclaration() methods.

The **enterTypeDeclaration()** method is called whenever the parser enters a class or interface declaration. This method adds import statements for the required libraries to the beginning of the file using the TokenStreamRewriter.

The **enterBlock()** method is called whenever the parser enters a block of code, such as a method body, if statement, or loop body. This method inserts code to log the visit of each block in the file using the TokenStreamRewriter.

After the parse tree has been walked, the modified token stream is converted back to text using the TokenStreamRewriter's getText() method, and the resulting text is written to a new file named **"Main.java"** using a FileWriter.

**Overall, this program demonstrates how ANTLR can be used to parse and modify source code files in a flexible and powerful way**.