**Integrating ANTLR with NetBeans 6.9**

ANTLR basically is a code generator. It takes grammar file as input and generates two classes : lexer and parser. Lexer runs first and splits input into pieces called tokens. Each token represents a meaningful piece of input.. The streams of tokens is passes to parser which do all necessary work. It builds AST (Abstract Syntax Tree), interpretes the code or translate it into some other form. Grammar file contains everything ANTLR needs to generate correct lexer and parser. Most importantly, grammar file describes how to split input into tokens and how to build tree from tokens. In other words, grammar file contains lexer rules and parser rules.

The following steps will tell you that how to integrate the ANTLR3 with the NetBeans GUI.

**Step 1**.

Download antlr-3.3-complete.jar from “antlr.org”.

**Step 2.**

Create a new NetBeans project and in it create a subdirectory. In that subdirectory create an empty file. In this empty file write the grammar containing lexer and parser rules and save the file with “.g” extension. The grammars files for our project “SQL query validation using ANTLR” are attached here too. You can also see them as references. Also write the main class having lexer and parser objects.

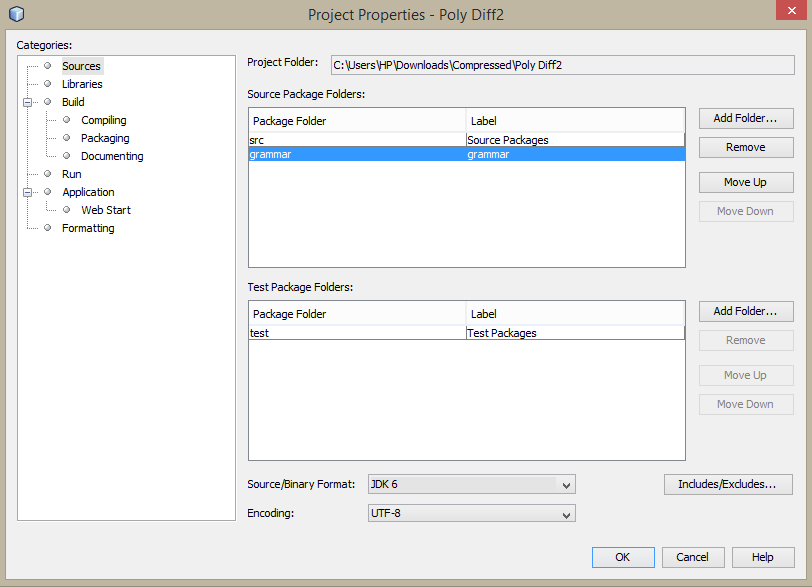
**Step 3.**

Tell the NetBeans IDE about the subdirectory you have created by following the steps shown below:

Project Properties Source Add Folder “Select the Subdirectory you have created”.

Also select “JDK 6” in “Source/ Binary format”.

At last click “OK” button.



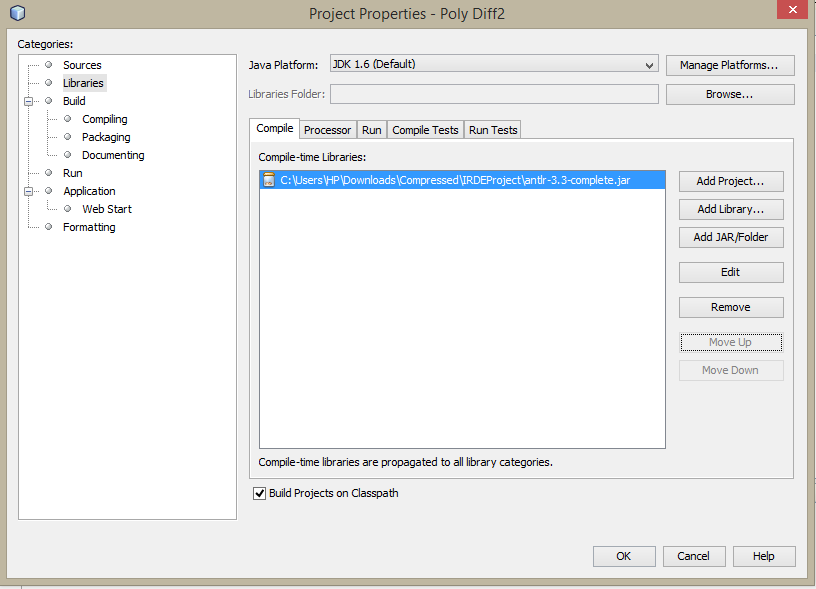
**Figure 1.1 Telling IDE about the created subdirectory**

**Step 4.**

In order to build an application using components defined in the grammar, the ANTLR compiler has to be invoked to generate Java code, before compilation with your other classes. So for it we have to tell the NetBeans IDE about the ANTLR jar. For that follow the procedure given:

In the Project Properties dialog, and the Libraries category, on the Compile tab, use the Add JAR/Folder button to add the ANTLR "complete" JAR to the compile-time libraries. You do this by navigating to the location where you have installed it. ( Mine is C:\Users\HP\Downloads\Compressed\IRDEProject\antlr-3.3-complete.jar.). i.e.

Project Properties Libraries Add JAR/Folder Add ANTLR jar location OK.



**Figure 1.2 Telling IDE about the ANTLR jar**

**Step 5.**

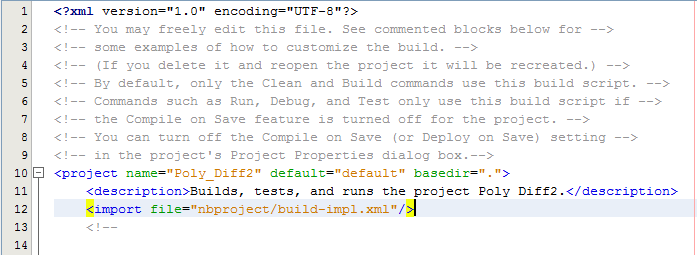
The next step will be adding a file. Add a new file named build-antlr-impl.xml in the “nbproject” subdirectory in the project. This is a prebuilt file which contains the logic of the compile and build processes invoked by the IDE. The contents of this file is shown below:



**Figure 1.3 Adding build-antlr-impl.xml file**

**Step 6.**

In the Files tab of the IDE navigate to the project directory and open the file build.xml which is a prebuilt file. This file controls the build, but by default it delegates every action to nbproject/build-impl.xml. The file begins like this:

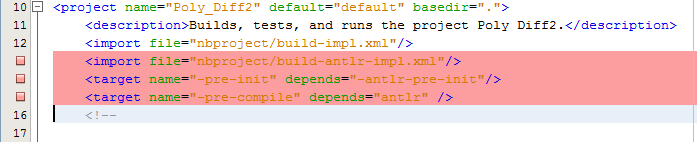


**Figure 1.4 The build.xml file**

We have to add extra 3 lines in this file to import the “build-antlr-impl.xml” file and its functions, we added in the previous step.

So add these lines and save the code.

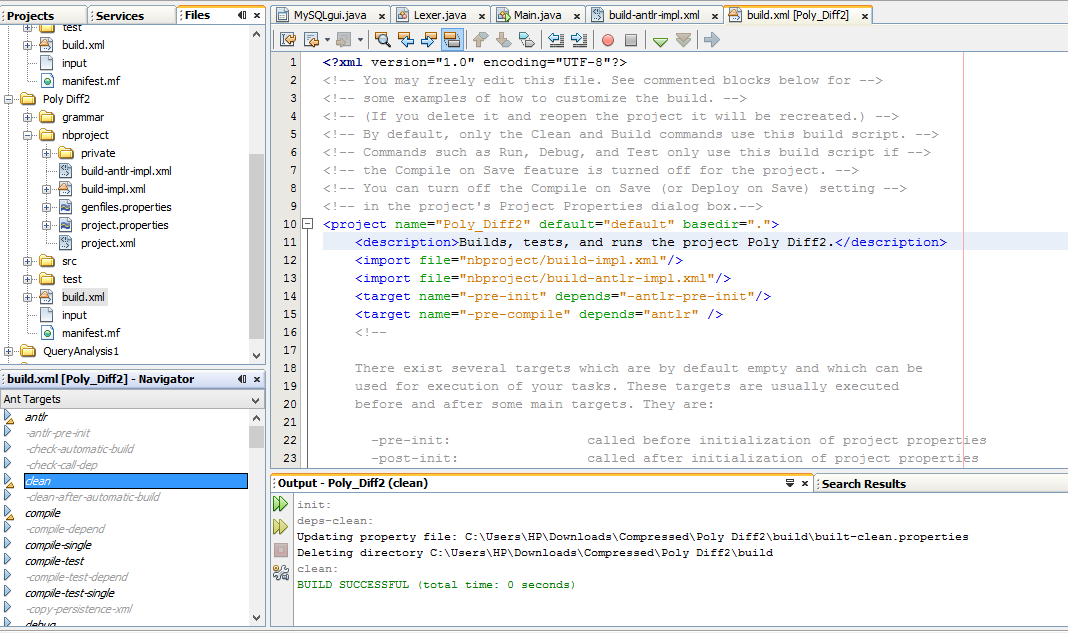
The three lines are highlighted in the figure below:



**Figure 1.5 The build.xml edited file**

**Step 7.**

Simply clean and build the application now. Give build.xml focus in the editor. The Navigator panel should show a series of Ant targets. Right-click the "clean" target and select Run Target. This just deletes any build directories for a clean start.



**Figure 1.6 Cleaning the project**

**Step 8.**

To build the project now, Right-click the "antlr" target and select Run Target. This runs the ANTLR compiler to generate the Java code for the Lexer, Parser, and other modules. Now a new source root has appeared in the Projects tab, and called "Generated Sources (antlr-output)". This packages includes the Java source files that are generated by ANTLR.

**Step 9.**

Now you can run the project and can take the input by creating an input text file and invoking it in main class or also you can make a GUI with netbeans and can take input from there too.

For further queries you can contact us:

Authors: ASMITA UNIYAL, asmitauniyal777@gmail.com

& ANUSHA GARG, anushagarg9@gmail.com