

Variant code handling (*conditional compiling*) for Java - simple approach:

Java does not support conditional compiling and it is very rare that you might need this. However, we did need it when migrating an application from JDK 1.4 to JDK 1.1.8 because of a PDA implementation.

USAGE:

This tool enables you to write one source file and include conditional compiling. For example, when you need to support different Java versions with one source code.

For each version, make a subdirectory (e.g. c:\mysource\1_4 and c:\mysource\1_1_8). Then start develop in one of those directories (let's assume in c:\mysource\1_4). Whenever you have some code which is only available on 1_1_8 then include it in special remarks (see below). After finishing the version 1_4, run this utility

e.g. `java srcVariants c:\mysource\1_4 c:\mysource\1_1_8 JDK_1_1_8`

Maybe later you have modified something in c:\mysource\1_1_8, then you can run the utility to copy files back:

e.g. `java srcVariants c:\mysource\1_1_8 c:\mysource\1_4 JDK_1_4`

The utility runs recursively deeper through the given source path (1st argument) and checks, if new or update files exist and processes them. Each file processed creates a backup file in the destination path with the extension .bak.

This only works on java sourcefiles with the extension .java.

Now, this utility reads your source file and "turns around" the remarks, based on this rules:

- A line looks like `///IFDEF xy` then "xy" is compared to the 3rd argument given to the program. If this matches, then the line is copied to the new created output file. If it is not matching a remark block is opened, starting with `///IFDEF xy`.
- A line looks like `///END_IFDEF xy` then "xy" is compared to the 3rd argument given to the program. If this matches, then the line is copied to the new created output file. If it is not matching a remark block is closed, starting with `///END_IFDEF xy`.
- During a remark block, normal remarks are switched to `///..* remark ..*` to avoid nested remarks.
- A line looks like `///IFDEF xy` then "xy" is compared to the 3rd argument given to the program. If this matches, then the line will be changed to `///IFDEF xy` and a subsequent remark is modified from `///..* remark ..*` entries to `///IFDEF remark IFDEF` entries.
- A line looks like `///END_IFDEF xy` then "xy" is compared to the 3rd argument given to the program. If this matches, then the line will be changed to `///END_IFDEF xy`.

Your source file might look like this:

```
// This is a code example
public class mysort
{
    // do some sorting, implementation now depends
    // on the JDK version
    public static void sort( String s[] )
    {
        if ( s != null )
        {
            ///IFDEF JDK_1_4
            Arrays.sort(s);
            ///END_IFDEF JDK_1_4

            ///IFDEF JDK_1_1_8
            for( int i = 0; i < s.length - 1; i++ )
                for( int j = i + 1; j < s.length; j++ )
                    if ( s[i].compareTo(s[j]) > 0 )
                    {
                        ///..* swap ..*
                        String h = s[i];
                        s[i] = s[j];
                        s[j] = h;
                    }
            ///END_IFDEF JDK_1_1_8*/
        }
    }
}
```

Note, that before or after these "//IFDEF xy" or /*IFDEF xy" you can use tabs or blanks, but not inside (means between "IFDEF" and "xy"). Only one tag per line.

You also can use more arguments rather than one version identifier following the 3rd argument.

Call the program without parameters to see how it is used. Before you use this utility, make sure to have a valid and current backup of your sourcefiles!

SUPPORT:

door2solution is not able to give you any support on this utility, but feel free to let us know your opinion about: office@door2solution.at

DISCLAIMER:

THIS UTILITY IS FREWARE AND FREE ADVICE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IF YOUR LOCALITY DOES NOT ALLOW THESE WARRANTY CONDITIONS, YOU ARE NOT GRANTED PERMISSION TO USE THIS UTILITY. DOOR2SOLUTION WILL NOT TAKE ANY RESPONSIBILITY OR LIABILITY FOR FAILURES, CORRECT, MODIFIED OR INCORRECT USAGE.

Advertising:

Ask for our [d2s.lib](#) for fast and easy development of web applications, PDA or local applications based on Java and JDBC databases.