Eclise JavaCC plugin

[JavaCC]

インストール

http://pagesperso-orange.fr/eclipse_javacc/

ここから、アーカイブをダウンロード、解凍して、Eclipseのフォルダに上書き。

プラグイン付属のテンプレート(サンプル)実行

コンテキストメニュー [New - Other] から、[JavaCC - JavaCC Templete File] を選択

Select a wizard	
Wizards: Type filter text	
type filter text	
Junit JavaCC Jin JavaCC JavaCC	
🕑 🦢 Java Emitter Templates	1
 JavaServer Faces JPA Mylyn 	
Plug=in Development Server SQL Development	-

JJ file を選択して次へ

IJ
Browse
Browse.
-

JavaCCSample Jee src Terror intotypea.sample.jav Terror java <new fil<="" th=""><th>acc.temptete</th><th>System.out.p System.out.p eg1.ReInit(S } catch (Error</th><th>rintln("NOK."); rintln(e.getMessage()); ystem.in); e} {</th></new>	acc.temptete	System.out.p System.out.p eg1.ReInit(S } catch (Error	rintln("NOK."); rintln(e.getMessage()); ystem.in); e} {
E Castomotontojon	New		<pre>> htln("Oops.");</pre>
 J. eg1 Token Manage J. Parse Exception.j. J. Simple CharStrea J. Token.java <new< li=""> J. Token.Mgr Error.ja </new<>	Open Open With Open Type Hierarchy Sho <u>w</u> In	F4 Alt+Shift+W	<pre>htln(e.getMessage()); </pre>
⊕ B new_file.jj	Copy	Ctrl+C	
	The Paste	Ctrl+V	
	💢 Delete	Delete	
	Armove from Context	Gtrl+Alt+Shift+Down	
	Build Path		•
41	Source	Alt+Shift+S	
ET LUX (Proving M)	Refactor	Alt+Shift+T	•
Task List	🚵 Import		Connol 9 Se Call Hi) & Ant 19
O IDV VERSION	🛃 Export_		VProgram Files¥, Java¥idk160.06¥t
P PARSER BEGIN	References		
E-t SKIP	Declarations		•
TOKEN			
E TOKEN	Hetresh	FD	
- r one_ine	Assign Working Sets		
-r term	<u>B</u> un As		•
-r unary	Debug As		🕨 💦 1 Debug on Server 🛛 Alt+Shift+D, R
r element	Profile As		🕨 🗊 2 Java Appl <mark>i</mark> cation 🛛 Alt+Shift+D, J
■ info.typea.sample.java	Validate Tgam		to Open Debug Dialog

コンソールから、数式を入力すると、解析され、構文がただしければ、OK、不正なら例外となる。

Reading	g from standa	rd in	but		
Enter (OK.	an expression	like	"1+(2+3)*4;"	:1+2+3;	

(1) 元になる、jj ファイル。ここに各種定義を書く。

(2) 上記 (1) のファイルから生成される Java クラス群

(3) jj ファイルの記述。PERSER_BEGIN(クラス名) から PERSER_END(クラス名) のクラス名部 分に、パーサーの処理を <u>Java</u> でコーディングする。

(4) パーサークラス名 (上記サンプルで実行したクラス 例:eg1)を指定。

(5) 構文解析の定義などを記述



本来なら、以下の手順を踏むところを、プラグインが一気に実行してくれるので非常に便利

1. javacc コマンド jj ファイル java ファイル群生成
 2. javac コマンド java ファイル群 class ファイル群生成
 3. java コマンド class ファイル 実行