

Sei folgende Grammatik für einen Ausschnitt der Java–Statements gegeben:

statement	\rightarrow	ifthenstatement ifthenelsestatement whilestatement block emptystatement returnstatement
ifthenstatement	\rightarrow	IF LBRACE bexpression RBRACE statement
ifthenelsestatement	\rightarrow	IF LBRACE bexpression RBRACE statement ELSE statement
whilestatement	\rightarrow	WHILE LBRACE bexpression RBRACE statement
block	\rightarrow	LBRACKET RBRACKET LBRACKET statements RBRACKET
statements	\rightarrow	statement statements statement
emptystatement	\rightarrow	SEMICOLON
returnstatement	\rightarrow	RETURN SEMICOLON RETURN expression SEMICOLON
bexpression	\rightarrow	TRUE
expression	\rightarrow	1

3.1 Java-Statement-LL-Parser

Schreiben Sie einen LL–Parser für die Menge Java–Statements. Nutzen Sie den Scanner aus Aufgabe 2.1 oder 2.2.

3.2 Java-Statement-Kombinatorparser

Schreiben Sie einen Kombinatorparser für die Menge Java–Statements, der die Eingabe in abstrakte Syntax umwandelt. Nutzen Sie den Scanner aus Aufgabe 2.1 oder 2.2.

3.3 Happy–Parser

Schreiben Sie einen Happy–Parser für die Menge Java–Statements, der die Eingabe in abstrakte Syntax umwandelt. Nutzen Sie den Scanner aus Aufgabe 2.1 oder 2.2.