# Konwerter okrojonej gramatyki języka Java na język C++ z uwzględnieniem prostych obiektów

## Spis tokenów

|  |  |
| --- | --- |
| **Identyfikator** - identyfikator składa się z jednej litery lub podkreślenia, po których może występować dowolna liczba liter, cyfr lub podkreśleń. | |
| Identifier | r’JavaLetter[JavaLetterOrDigit]\*’ |
| JavaLetter | r’[a-zA-Z\_]’ |
| JavaLetterOrDigit | r’[a-zA-Z0-9\_]’ |
|  | |
| **Słowa kluczowe** | |
| BOOLEAN | ‘boolean’ |
| BREAK | ‘break’ |
| CASE | ‘case’ |
| CHAR | ‘char’ |
| CLASS | ‘class’ |
| CONST | ‘const’ |
| CONTINUE | ‘continue’ |
| DOUBLE | ‘double’ |
| ELSE | ‘else’ |
| ENUM | ‘enum’ |
| FLOAT | ‘float’ |
| FOR | ‘for’ |
| IF | ‘if’ |
| INT | ‘int’ |
| PRIVATE | ‘private’ |
| PROTECTED | ‘protected’ |
| PUBLIC | ‘public’ |
| RETURN | ‘return’ |
| SWITCH | ‘switch’ |
| WHILE | ‘while’ |
|  | |
| **Literały** | |
| Literal | r’[IntegerLiteral|FloatingPointLiteral|BooleanLiteral|CharacterLiteral| NullLiteral]’ |
| IntegerLiteral | r’-?[1-9][0-9]\*’ |
| FloatingPointLiteral | r’-?([0-9]+[.][0-9]\*|[.][0-9]+)([eE][-+]?[0-9]+)?’ |
| BooleanLiteral | r’[True|False]’ |
| CharacterLiteral | r'([^'\\]|\\['"\\bfnrt])\*' |
| NullLiteral | ‘null’ |
|  | |
| **Produkcje? (typy, wartości, zmienne)** | |
| primitiveType | r‘[numericType|boolean]’ |
| numericType | r‘’[integralType|floatingPointType] |
| integralType | r‘[int|char]’ |
| floatingPointType | ‘float’ |
|  | |
| referenceType | r‘[typeVariable|arrayType  ]’ |