```java
package Lexer;

/**
 * Created by rubenspessoa on 04/09/16.
 */
public class Token {

    /**
     * Internal representation of Tokens.
     */
    public enum TokenCategory {

        EOF(-1),
        ID(1),
        CTE_FLOAT(2),
        CTE_INT(3),
        CTE_STR(4),
        OP_ATR(5),
        OP_REL1(6),
        OP_REL2(7),
        OP_AD(8),
        OP_MULT(9),
        COMMENT(10),
        VECTOR_AUX(11),
        OP_AND(12),
        OP_OR(13),
        OP_NOT(14),
        PR_IF(15),
        PR_ELSE(16),
        PR_SHOOT(17),
        PR_WHILE(18),
        PR_FOR(19),
        TYPE_VALUE(20),
        BOOL_VALUE(21),
        SP(22),
        AB_PAR(23),
        FEC_PAR(24),
        AB_CH(25),
        FEC_CH(26),
        AB_COL(27),
        FEC_COL(28),
        PR_VOID(29),
        PR_IO(30),
        PR_MAIN(31);

        private int value;

        TokenCategory(int value) {
            this.value = value;
        }
        public int getValue() {
            return this.value;
        }
    }


    private TokenCategory tokenCategory;
    private String sequence;
    private int lin, col;
```

```java
/**
 * Construct the token representation within its values
 * @param tokenCategory Category Identifier int
 * @param sequence String that represents the token
 * @param lin Lexer.Token line in input
 * @param col Lexer.Token column in input
 */

public Token(TokenCategory tokenCategory,
             String sequence,
             int lin,
             int col) {

    this.tokenCategory = tokenCategory;
    this.sequence = sequence;
    this.lin = lin;
    this.col = col;
}

@Override
public String toString() {
    return "(Sequence: " + sequence + ", Token Category: " +
        tokenCategory.getValue() + ", Position: (" + this.lin + ", " +
        this.col + "))\n";
}

public TokenCategory getTokenCategory() {
    return this.tokenCategory;
}
}
```