Features of Syntactic Constructions

PROGRAM

Name	Description	Domain
Class percentage	Percentage of classes (out of all the types) defined in that program.	[0, 100]
Code in default package	Whether the program implements types in the default package.	True or False.
Code in packages	Whether the program implements types inside packages.	True or False.
Enum percentage	Percentage of enumerations (out of all the types) defined in that program.	[0, 100]
Interface percentage	Percentage of interfaces (out of all the types) defined in that program.	[0, 100]
NumberOfTypesInDefaultPackage	Number of Types in the Default Package	[0,n]
NumberOfTypesInPackages	Number of Types in packages	[0,n]
User ID	Identification of the student using the subject and a number assigned.	String

TYPE DEFINITION

Name	Description	Domain
Abstract	Defines if the type definition is abstract or not	True or False.
File ID	File this type definition belongs.	[0, n]
Final	Whether or not the type is final.	True or False.

Has Extends	Whether the type extends another type.	True or False.
Inner Class	Defines if it is an inner class or not.	True or False.
IsInDefaultPackage	Defines if the type definition is the default package or not	True or False.
Naming convention	Naming convention used for the type name.	Lower, Upper, Camel_Low, Camel_Up, Snake_Case or Anonymous
Nested Class	Defines if it is a nested class or not.	True or False.
Node Type	Syntactic category of the current node, given the abstract grammar for the Java language.	Class, Interface or Enumeration.
Number of annotations	Number of annotations declared for that type.	[0, n]
Number of constructors	Number of constructors implemented in that type.	[0, n]
Number of fields	Number of fields defined in that type.	[0, n]
Number of generics	Number of generic types declared for that type.	[0, n]
Number of implements	Number of interfaces implemented.	[0, n]
Number of Inner Types	Number of Inner Types. (An inner class is nested but non-static)	[0, n]
Number of methods	Number of methods declared for that type.	[0, n]
Number of Nested Types	Number of Nested Types.	[0, n]
Number of Static Blocks	Number of statics types declared for that type.	[0, n]
Number of Static Nested Types	Number of Static nested types.	[0, n]
Percentage of overloaded methods	Percentage of overloaded methods declared for that type.	[0, 100]

Public Visibility	Whether or not the type visibility is public or not.	True or False.
Static	Whether or not the type is static.	True or False.
Static Field Percentage	Percentage of fields that are static.	[0, 100]
Static Method Percentage	Percentage of methods that are static.	[0, 100]
Strict TFP	Defines if it is a StrictFTP type definition or not.	True or False.

FIELD DEFINITION

Name	Description	Domain
Final	Whether or not the field is final	True or False.
Initial Value	Initial value, if exists.	Expression[3] | None
Naming convention	Naming convention used for the field name.	Lower, Upper, Camel_Low, Camel_Up, Snake_Case
Number of annotations	Number of annotations declared for that field.	[0,n]
Parent Node	Syntactic category of the parent node, given the abstract grammar for the Java language.	Class, Interface or Enumeration.
SourceCode	Source code	JavaCode
Static	True or False.	True or False.
Transient	Indicates if the field is transient or not	True of False.
Type	Field type	Type [6]
Visibility	The visibility of the field.	Public, Protected, Package or Private.
Volatile	Indicates if the field is volatile or not	True of False.

METHOD DEFINITION

Name	Description	Domain
Abstract	Defines if the method is abstract or not.	True or False.
Constructor	Defines if the method is a constructor or not.	True or False.
Default Implementation	Defines if the method uses a default implementation.	True of False.
Final	Defines if the method is final or not.	True of False.
First, second and third parameter	Type of the corresponding method parameter.	Type [6] | None
First, second and third parameter ID	Type ID of the corresponding parameter.	[0, n]
Has Overrides	Defines if the method has overrides or not.	True of False.
Main naming locals	Main naming convention used for local variables.	Lower, Upper, Camel_Low, Camel_Up or Snake_Case
Naming convention	Naming convention used for the method name.	Lower, Upper, Camel_Low, Camel_Up or Snake_Case | Constructor
Native	Defines if the method is native or not.	True or False.
Number of annotations	Number of annotations declared for that method.	[0, n]
Number of generics	Number of generic types declared for that method.	[0, n]
Number of Inner Classes	Number of inner classes declared.	[0, n]
Number of local variables	Number of local variables declared.	[0, n]
Number of Overloaded Methods	Number of overloaded methods found for that method name.	[0, n]

Number of parameters	Number of declared parameters.	[0, n]
Number of statements	Number of statements used in the method implementation.	[0, n]
Number of throws	Number of exceptions declared in the throw clause.	[0, n]
Return Type	Method Return Type.	Type [6] | Void | Constructor
Return type ID	Return Type ID	[0, n]
SourceCode	Source code	JavaCode (signatura QualifiedName)
Static	Defines if the method is static or not.	True of False.
StrictTFP	Defines if the method is StrictFTP or not.	True or False.
Synchronized	Defines if the method is synchronized or not.	True or False.
Visibility	The visibility of the method.	Public, Protected, Package or Private.

STATEMENT

Name	Description	Domain
Category	Syntactic category of the current node, given the abstract grammar for the Java language.	Statement[1]
Depth	Maximum distance (number of edges) of the longest path from the current node to a leaf node.	[0, n]
First, second and third child	Syntactic category of the corresponding child node.	Statement[1] | Expression[3] | Type [6] | None
First, second and third child ID	Child ID of the corresponding parameter.	[0,n]
Height	Distance (number of edges) from the current node to the root node in the enclosing type (class, interface or enumeration).	[0, n]

Parent node	Syntactic category of the parent node	Statement[1] | Expression[3*] | MethodDefinition
Role	Role played by the current node in the structure of its parent node.	StatementRole[2]
SourceCode	Source code	JavaCode (toda la sentencia)

EXPRESSION

Name	Description	Domain
Category	Syntactic category of the current node.	Expression[3]
Depth	Maximum distance (number of edges) of the longest path from the current node to a leaf node.	[0, n]
First, second and third child	Syntactic category of the corresponding child node.	Statement[1] | Expression[3] | Type[6] | None
First, second and third child ID	Child ID of the corresponding parameter.	[0,n]
Height	Distance (number of edges) from the current node to the root node in the enclosing type (class, interface or enumeration)	[0, n]
Parent node	Syntactic category of the parent node.	Statement[1] | Expression[3] | FieldDefinition
Role	Role played by the current node in the structure of its parent node	ExpressionRole[4]

TYPE

Name	Description	Domain
Category	Syntactic category of the current node.	JavaType[3]
Height	Distance (number of edges) from the current node to the	[0, n]

	root node in the enclosing type.	
Number of Dimensions	Number of dimensions of the declared variable.	[0, n]
Number of Generics	Number of generics declared in the definition.	[0, n]
Parent Node	Syntactic category of the parent node.	FieldDefinition | MethodDefinition | VariableDefinition | CastExpression | InstanceOf | NewArray | NewClass | NewClassStatement | MethodInvocation | MethodInvocationStatement
Primitive	Whether or not the type is primitive.	True or False.
Role	Role played by the current node in the structure of its parent node.	TypeRole[7]

[1] Statement

Assert, AssignmentStatement, Break, Case, Catch, CompundAssignmentStatement, Continue, DoWhileLoop, EmptyStatement, EnhancedForLoop, ForLoop, If, Label, MethodInvocationStatement, NewClassStatement, PostfixDecrementUnaryStatement, PostfixIncrementUnaryStatement, PrefixDecrementUnaryStatement, PrefixIncrementUnaryStatement, Return, StaticBlock, Switch, Synchronized, Throw, Try, VariableDefinition, WhileLoop

[2] StatementRole

CatchBodyStatement, CatchParameters, CatchStatement, DoWhileBodyStatement, EnhancedForLoopBodyStatement, EnhancedForLoopVariable, FinallyBodyStatement, ForLoopBodyStatement, ForLoopInit, ForLoopUpdate, IfElseStatement, IfThenStatement, LabelBodyStatement, LambdaBodyStatement, LambdaParameterStatement, MethodBodyStatement, MethodParameterStatement, NewClassBody, StaticBlockConstructor, SwitchCaseStatement, SynchronizedBodyStatement, TryBodyStatement, TryResource, VariableDeclarationModifiers, WhileBodyStatement

[3] Expression

Annotation, ArithmeticBinary, ArithmeticUnary, ArrayAccess, Assignment, BitwiseBinary, BitwiseUnary, BooleanLiteral, CastExpression, CharLiteral, ComparisonBinary,

CompundAssignment, ConditionalExpression, DoubleLiteral, FloatLiteral, InstanceOf, IntLiteral, LambdaExpression, LambdaExpressionBody, LambdaExpressionParameter, LogicalBinary, LogicalUnary, LongLiteral, MemberReference, MemberSelect, MethodInvocation, NewArray, NewClass, NullLiteral, PostfixDecrementUnary, PostfixIncrementUnary, PrefixDecrementUnary, PrefixIncrementUnary, ShortLiteral, StringLiteral, Variable

4 ExpressionRole

AnnotationsArguments, ArrayAccessExpression, ArrayAccessIndex, AssertCondition, AssertDetail, AssigmentLHS, AssigmentRHS, BinopLHS, BinopRHS, CastEncloses, CompundAssignmentExpression, CompundAssignmentVariable, ConditionalCondition, ConditionalElse, ConditionalThen, DoWhileCondition, EnhancedForLoopExpression, FieldAnnotation, ForLoopCondition, IfCondition, InstanceOfExpression, LambdaBody, LambdaParameter, MemberReferenceQualifierExpression, MemberSelectedExpression, MethodAnnotation, MethodInvocationArguments, MethodInvocationMethodSelect, NewArrayDimension, NewArrayInitializer, NewClassArguments, NewClassBody, NewClassEnclosingExpression, ReturnExpression, SwitchCaseExpression, SwitchExpression, SynchronizedExpression, ThrowExpression, TypeDefinitionAnnotation, UnaryEncloses, VariableDeclarationInit, WhileCondition

5 JavaType

Boolean, Byte, Char, Double, Float, Int, IntersectionType, Long, Reference, Short, String, UnionType

6 Type

Array, Boolean, Byte, Char, Double, Float, Generic, Int, IntersectionType, Long, Reference, Short, String, UnionType, Void

7 TypeRole

CastType, FieldType, InstanceOfType, NewArrayType, NewClassType, ParameterType, ReturnType, VariableType