

# Syntax generated by **parseYOut**

```
<OdISpecification> ::= <Definition> .
                    | <Definition> ;
                    | <Definition> ; <OdISpecification>

<Definition> ::= <TypeDcl>
                | <ConstDcl>
                | <ExceptDcl>
                | <Interface>
                | <RuleDcl>
                | <ExtRuleDcl>
                | <ThesaurusRelation>
                | <Module>
                | <error>

<TypeDcl> ::= typedef <TypeDeclarator>
           | <ConstrTypeSpec>

<TypeDeclarator> ::= <TypeSpec> <Declarators>
<Declarators> ::= <Declarator>
                 | <Declarators> , <Declarator>

<Declarator> ::= <Identifier>
                | <Identifier> <ArraySizeList>

<ArraySizeList> ::= <FixedSize>
                   | <ArraySizeList> <FixedSize>

<TypeSpec> ::= <SimpleTypeSpec>
              | <ConstrTypeSpec>

<SimpleTypeSpec> ::= <BaseTypeSpec>
                   | <TemplateTypeSpec>
                   | <Identifier>

<BaseTypeSpec> ::= <FloatingPtType>
                  | <IntegerType>
                  | <CharType>
                  | <BooleanType>
                  | <OctetType>
                  | <RangeType>
                  | <DateType>
                  | <AnyType>
```

⟨FloatingPtType⟩	::=	<b>float</b>
		<b>double</b>
⟨DateType⟩	::=	⟨DATE⟩
		⟨TIMESTAMP⟩
⟨IntegerType⟩	::=	⟨LongIntType⟩
		<b>short</b>
		<b>unsigned long</b>
		<b>unsigned short</b>
⟨LongIntType⟩	::=	<b>integer</b>
		<b>int</b>
		<b>long</b>
⟨CharType⟩	::=	<b>char</b>
⟨BooleanType⟩	::=	<b>boolean</b>
⟨OctetType⟩	::=	<b>octet</b>
⟨RangeType⟩	::=	<b>range</b> { ⟨RangeSpecifier⟩ }
⟨RangeSpecifier⟩	::=	⟨SignedIntegerLiteral⟩ , ⟨SignedIntegerLiteral⟩
		⟨SignedIntegerLiteral⟩ , + <b>infinite</b>
		- <b>infinite</b> , ⟨SignedIntegerLiteral⟩
⟨AnyType⟩	::=	<b>any</b>
⟨TemplateTypeSpec⟩	::=	⟨ArrayType⟩
		⟨StringType⟩
		⟨CollectionType⟩
⟨ArrayType⟩	::=	<b>array</b> < ⟨SimpleTypeSpec⟩ , ⟨IntegerLiteral⟩ >
⟨\$\$1⟩	::=	
⟨ArrayType⟩	::=	<b>array</b> < ⟨SimpleTypeSpec⟩ > ⟨\$\$1⟩ <b>sequence</b> < ⟨SimpleTypeSpec⟩ , ⟨IntegerLiteral⟩ >
⟨StringType⟩	::=	<b>string</b> [ ⟨IntegerLiteral⟩ ]
		<b>string</b>
⟨CollectionType⟩	::=	⟨AttrCollectionSpecifier⟩ < ⟨SimpleTypeSpec⟩ >
⟨AttrCollectionSpecifier⟩	::=	<b>set</b>
		<b>list</b>
		<b>bag</b>
⟨ConstrTypeSpec⟩	::=	⟨StructType⟩
		⟨UnionType⟩
		⟨EnumType⟩
⟨StructType⟩	::=	<b>struct</b> ⟨Identifier⟩ { ⟨MemberList⟩ }
⟨MemberList⟩	::=	⟨Member⟩
		⟨MemberList⟩ ⟨Member⟩

⟨Member⟩	::=	⟨TypeSpec⟩ ⟨Declarators⟩ ;
⟨UnionType⟩	::=	<b>union</b> ⟨Identifier⟩ <b>switch</b> ( ⟨SwitchTypeSpec⟩ ) { ⟨SwitchBody⟩ }
⟨SwitchTypeSpec⟩	::=	⟨IntegerType⟩   ⟨CharType⟩   ⟨BooleanType⟩   ⟨EnumType⟩   ⟨ScopedName⟩   ⟨RangeType⟩
⟨SwitchBody⟩	::=	⟨Case⟩   ⟨Case⟩ ⟨SwitchBody⟩
⟨Case⟩	::=	⟨CaseLabelList⟩ ⟨ElementSpec⟩ ;
⟨CaseLabelList⟩	::=	⟨CaseLabel⟩   ⟨CaseLabel⟩ ⟨CaseLabelList⟩
⟨CaseLabel⟩	::=	<b>case</b> ⟨ConstExp⟩ :   <b>default</b> :
⟨ElementSpec⟩	::=	⟨TypeSpec⟩ ⟨Declarator⟩
⟨EnumType⟩	::=	<b>enum</b> ⟨Identifier⟩ { ⟨EnumeratorList⟩ }
⟨EnumeratorList⟩	::=	⟨Enumerator⟩   ⟨EnumeratorList⟩ , ⟨Enumerator⟩
⟨Enumerator⟩	::=	⟨Identifier⟩
⟨ConstExp⟩	::=	⟨OrExpr⟩
⟨OrExpr⟩	::=	⟨XOrExpr⟩   ⟨OrExpr⟩   ⟨XOrExpr⟩
⟨XOrExpr⟩	::=	⟨AndExpr⟩   ⟨XOrExpr⟩ ^ ⟨AndExpr⟩
⟨AndExpr⟩	::=	⟨ShiftExpr⟩   ⟨AndExpr⟩ & ⟨ShiftExpr⟩
⟨ShiftExpr⟩	::=	⟨AddExpr⟩   ⟨ShiftExpr⟩ >> ⟨AddExpr⟩   ⟨ShiftExpr⟩ << ⟨AddExpr⟩
⟨AddExpr⟩	::=	⟨MultExpr⟩   ⟨AddExpr⟩ + ⟨MultExpr⟩   ⟨AddExpr⟩ - ⟨MultExpr⟩

⟨MultExpr⟩	::=	⟨UnaryExpr⟩
		⟨MultExpr⟩ * ⟨UnaryExpr⟩
		⟨MultExpr⟩ / ⟨UnaryExpr⟩
		⟨MultExpr⟩ \% ⟨UnaryExpr⟩
⟨UnaryExpr⟩	::=	⟨Signes⟩ ⟨PrimaryExpr⟩
		⟨PrimaryExpr⟩
⟨PrimaryExpr⟩	::=	⟨Identifier⟩
		⟨IntegerLiteral⟩
		⟨FloatingPtLiteral⟩
		( ⟨OrExpr⟩ )
		( ⟨error⟩ )
⟨ConstDcl⟩	::=	<b>const</b> ⟨StringType⟩ ⟨Identifier⟩ = ⟨StringLiteral⟩
		<b>const</b> ⟨CharType⟩ ⟨Identifier⟩ = ⟨CharacterLiteral⟩
		<b>const</b> ⟨IntegerType⟩ ⟨Identifier⟩ = ⟨ConstExp⟩
		<b>const</b> ⟨FloatingPtType⟩ ⟨Identifier⟩ = ⟨ConstExp⟩
⟨SignedIntegerLiteral⟩	::=	⟨IntegerLiteral⟩
		⟨Signes⟩ ⟨IntegerLiteral⟩
⟨SignedFloatingPtLiteral⟩	::=	⟨Signes⟩ ⟨FloatingPtLiteral⟩
		⟨FloatingPtLiteral⟩
⟨Signes⟩	::=	-
		+
⟨ExceptDcl⟩	::=	<b>exception</b> ⟨Identifier⟩ { ⟨OptMemberList⟩ }
⟨OptMemberList⟩	::=	
		⟨MemberList⟩
⟨ScopedName⟩	::=	⟨Identifier⟩
		:: ⟨Identifier⟩
		⟨ScopedName⟩ :: ⟨Identifier⟩
⟨Interface⟩	::=	⟨InterfaceDcl⟩
⟨Int View⟩	::=	<b>interface</b>
		<b>view</b>

⟨InterfaceDcl⟩	::=	⟨IntView⟩ ⟨Identifier⟩ : ⟨InheritanceSpec⟩ ⟨OptTypePropertyList⟩ ⟨OptPersistenceDcl⟩ ⟨SingleInterfaceBody⟩   ⟨IntView⟩ ⟨Identifier⟩ : ⟨InheritanceSpec⟩ ⟨OptTypePropertyList⟩ ⟨OptPersistenceDcl⟩ ⟨SingleInterfaceBody⟩ ⟨InterfaceBodyUnionList⟩   ⟨IntView⟩ ⟨Identifier⟩ ⟨OptTypePropertyList⟩ ⟨OptPersistenceDcl⟩ ⟨SingleInterfaceBody⟩   ⟨IntView⟩ ⟨Identifier⟩ ⟨OptTypePropertyList⟩ ⟨OptPersistenceDcl⟩ ⟨SingleInterfaceBody⟩ ⟨InterfaceBodyUnionList⟩
⟨InheritanceSpec⟩	::=	⟨Identifier⟩   ⟨InheritanceSpec⟩ , ⟨Identifier⟩
⟨OptTypePropertyList⟩	::=	( ⟨OptSourceSpec⟩ ⟨OptExtentSpec⟩ ⟨OptKeySpec⟩ ⟨OptCandKeySpec⟩ ⟨OptForKeySpec⟩ )
⟨OptSourceSpec⟩	::=	<b>source</b> ⟨SourceType⟩ ⟨Identifier⟩
⟨SourceType⟩	::=	<b>relational</b>   <b>nfrelational</b>   <b>object</b>   <b>file</b>   <b>semistructured</b>
⟨OptExtentSpec⟩	::=	<b>extent</b> ⟨ListExtent⟩
⟨ListExtent⟩	::=	⟨Identifier⟩   ⟨ListExtent⟩ , ⟨Identifier⟩
⟨OptKeySpec⟩	::=	<b>key</b> ⟨Key⟩
⟨OptCandKeySpec⟩	::=	⟨CandKeySpecList⟩
⟨CandKeySpecList⟩	::=	⟨CandKeySpec⟩   ⟨CandKeySpecList⟩ ⟨CandKeySpec⟩
⟨CandKeySpec⟩	::=	<b>candidate_key</b> ⟨Identifier⟩ ⟨Key⟩
⟨OptForKeySpec⟩	::=	⟨ForKeySpecList⟩
⟨ForKeySpecList⟩	::=	⟨ForKeySpec⟩   ⟨ForKeySpecList⟩ ⟨ForKeySpec⟩
⟨ForKeySpec⟩	::=	<b>foreign_key</b> ( ⟨ForeignKeyList⟩ ) <b>references</b> ⟨Identifier⟩ ⟨OptRefKeyList⟩

⟨OptRefKeyList⟩	::=	( ⟨ForeignKeyList⟩ )
⟨ForeignKeyList⟩	::=	⟨Identifier⟩
		⟨ForeignKeyList⟩ , ⟨Identifier⟩
⟨Key⟩	::=	( ⟨PropertyList⟩ )
⟨PropertyList⟩	::=	⟨PropertyName⟩
		⟨PropertyList⟩ , ⟨PropertyName⟩
⟨PropertyName⟩	::=	⟨Identifier⟩
⟨OptPersistenceDcl⟩	::=	
		<b>persistent</b>
		<b>transient</b>
⟨InterfaceBodyUnionList⟩	::=	⟨InterfaceBodyUnion⟩
		⟨InterfaceBodyUnionList⟩ ⟨InterfaceBodyUnion⟩
⟨InterfaceBodyUnion⟩	::=	<b>union</b> ⟨Identifier⟩ ⟨SingleInterfaceBody⟩
⟨SingleInterfaceBody⟩	::=	{ ⟨InterfaceBody⟩ }
⟨InterfaceBody⟩	::=	⟨Export⟩ ;
		⟨Export⟩ ; ⟨InterfaceBody⟩
⟨Export⟩	::=	⟨TypeDcl⟩
		⟨ConstDcl⟩
		⟨ExceptDcl⟩
		⟨AttrDcl⟩
		⟨RelDcl⟩
		⟨OpDcl⟩
⟨AttrDcl⟩	::=	⟨OptReadOnly⟩ <b>attribute</b> ⟨DomainType⟩ ⟨AttributeName⟩
		⟨OptFixedSize⟩ ⟨OptMappingRuleDcl⟩
⟨OptReadOnly⟩	::=	
		<b>readonly</b>
⟨AttributeName⟩	::=	⟨Identifier⟩
		⟨Identifier⟩ ?
⟨DomainType⟩	::=	⟨SimpleTypeSpec⟩
		⟨StructType⟩
		⟨EnumType⟩
⟨OptFixedSize⟩	::=	
		⟨FixedSize⟩
⟨FixedSize⟩	::=	[ ⟨IntegerLiteral⟩ ]
⟨OptMappingRuleDcl⟩	::=	
		⟨MappingRuleDcl⟩
⟨MappingRuleDcl⟩	::=	<b>mapping rule</b> ⟨MapRuleList⟩

⟨MapRuleList⟩	::=	⟨MapRule⟩   ⟨MapRuleList⟩ , ⟨MapRule⟩
⟨MapRule⟩	::=	⟨LocalAttributeName⟩   ⟨DefaultValue⟩   ⟨MapAndExpression⟩   ⟨MapUnionExpression⟩
⟨LocalAttributeName⟩	::=	⟨LocalClassName⟩ . ⟨Identifier⟩
⟨LocalClassName⟩	::=	⟨Identifier⟩ . ⟨Identifier⟩
⟨DefaultValue⟩	::=	⟨Identifier⟩ . ⟨Identifier⟩ = ⟨StringLiteral⟩
⟨MapAndExpression⟩	::=	( ⟨MapAndList⟩ <b>and</b> ⟨LocalAttributeName⟩ )
⟨MapAndList⟩	::=	⟨LocalAttributeName⟩   ⟨MapAndList⟩ <b>and</b> ⟨LocalAttributeName⟩
⟨MapUnionExpression⟩	::=	( ⟨MapUnionList⟩ <b>union</b> ⟨LocalAttributeName⟩ <b>on</b> ⟨Identifier⟩ )
⟨MapUnionList⟩	::=	⟨LocalAttributeName⟩   ⟨MapUnionList⟩ <b>union</b> ⟨LocalAttributeName⟩
⟨RelDcl⟩	::=	<b>relationship</b> ⟨TargetOfPath⟩ ⟨Identifier⟩ <b>inverse</b> ⟨InverseTraversalPath⟩ ⟨OptOrderBy⟩
⟨TargetOfPath⟩	::=	⟨Identifier⟩   ⟨RelCollectionType⟩ < ⟨Identifier⟩ >
⟨RelCollectionType⟩	::=	<b>set</b>   <b>list</b>
⟨InverseTraversalPath⟩	::=	⟨Identifier⟩ :: ⟨Identifier⟩
⟨OptOrderBy⟩	::=	{ <b>order_by</b> ⟨ScopedNameList⟩ }
⟨ScopedNameList⟩	::=	⟨ScopedName⟩   ⟨ScopedNameList⟩ , ⟨ScopedName⟩
⟨OpDcl⟩	::=	⟨OptOneway⟩ ⟨OperTypeSpec⟩ ⟨Identifier⟩ ⟨ParameterDcls⟩ ⟨OptRaisesExpr⟩ ⟨OptContextExpr⟩
⟨OptOneway⟩	::=	<b>oneway</b>
⟨OperTypeSpec⟩	::=	⟨SimpleTypeSpec⟩   <b>void</b>
⟨ParameterDcls⟩	::=	( ⟨ParamDclList⟩ )   ( )
⟨ParamDclList⟩	::=	⟨ParamDcl⟩   ⟨ParamDclList⟩ , ⟨ParamDcl⟩
⟨ParamDcl⟩	::=	⟨ParamAttribute⟩ ⟨SimpleTypeSpec⟩ ⟨Declarator⟩

⟨ParamAttribute⟩	::=		<b>in</b>
			<b>out</b>
			<b>inout</b>
⟨OptRaisesExpr⟩	::=		<b>raises</b> ( ⟨ScopedNameList⟩ )
⟨OptContextExpr⟩	::=		<b>context</b> ( ⟨StringLiteralList⟩ )
⟨StringLiteralList⟩	::=		⟨StringLiteral⟩
			⟨StringLiteralList⟩ , ⟨StringLiteral⟩
⟨ExtRuleDcl⟩	::=		<b>extrule</b> ⟨Identifier⟩ ⟨ExtRuleSpec⟩
⟨ExtRuleSpec⟩	::=		⟨ForAll⟩ ⟨Identifier⟩ <b>in</b> ⟨ExtRuleType⟩
⟨ExtRuleType⟩	::=		⟨ExtRuleIsa⟩
			⟨ExtRuleBottom⟩
⟨ExtRuleBottom⟩	::=		( ⟨LocalClassName⟩ <b>and</b> ⟨LocalClassName⟩ ) <b>then</b>
			⟨Identifier⟩ <b>in bottom</b>
⟨ExtRuleIsa⟩	::=		⟨LocalClassName⟩ <b>then</b> ⟨Identifier⟩ <b>in</b> ⟨LocalClassName⟩
⟨RuleDcl⟩	::=		<b>rule</b> ⟨Identifier⟩ ⟨RuleSpec⟩
⟨RuleSpec⟩	::=		⟨ForAll⟩ ⟨Identifier⟩ <b>in</b> ⟨Identifier⟩ : ⟨RuleBodyList⟩ <b>then</b>
			⟨RuleBodyList⟩
			{ <b>case of</b> ⟨Identifier⟩ : ⟨CaseList⟩ }
⟨ForAll⟩	::=		<b>for all</b>
			<b>forall</b>
⟨RuleBodyList⟩	::=		( ⟨RuleBodyList⟩ )
			⟨RuleBody⟩
			⟨RuleBodyList⟩ <b>and</b> ⟨RuleBody⟩
⟨\$\$2⟩	::=		
⟨RuleBody⟩	::=		⟨DottedName⟩ ⟨RuleConstOp⟩ ⟨OptRuleCast⟩
			⟨LiteralValue⟩ ⟨\$\$2⟩ ⟨DottedName⟩ ⟨RuleConstOp⟩
			⟨OptRuleCast⟩ ⟨DottedName⟩
			⟨DottedName⟩ <b>in</b> ⟨DottedName⟩
			⟨ForAll⟩ ⟨Identifier⟩ <b>in</b> ⟨DottedName⟩ : ⟨RuleBodyList⟩
			<b>exists</b> ⟨Identifier⟩ <b>in</b> ⟨DottedName⟩ : ⟨RuleBodyList⟩
			⟨DottedName⟩ = ⟨SimpleTypeSpec⟩ ⟨Identifier⟩ (
			⟨DottedLiteralList⟩ )
⟨DottedLiteralList⟩	::=		⟨DottedLiteral⟩
			⟨DottedLiteralList⟩ , ⟨DottedLiteral⟩



⟨DottedLiteral⟩	::=	⟨OptSimpleTypeSpec⟩ ⟨DottedName⟩   ⟨OptSimpleTypeSpec⟩ ⟨LiteralValue⟩
⟨OptSimpleTypeSpec⟩	::=	⟨SimpleTypeSpec⟩
⟨RuleConstOp⟩	::=	=   >=   <=   <   >
⟨LiteralValue⟩	::=	⟨SignedFloatingPtLiteral⟩   ⟨SignedIntegerLiteral⟩   ⟨CharacterLiteral⟩   ⟨StringLiteral⟩
⟨DottedName⟩	::=	⟨Identifier⟩   ⟨DottedName⟩ . ⟨Identifier⟩
⟨OptRuleCast⟩	::=	( ⟨SimpleTypeSpec⟩ )
⟨CaseList⟩	::=	⟨CaseSpec⟩   ⟨CaseList⟩ ⟨CaseSpec⟩
⟨CaseSpec⟩	::=	⟨LiteralValue⟩ : ⟨DottedName⟩
⟨ThesaurusRelation⟩	::=	⟨LocalAttributeName⟩ ⟨ThesRelType⟩ ⟨LocalAttributeName⟩   ⟨LocalClassName⟩ ⟨ThesRelType⟩ ⟨LocalClassName⟩   ⟨LocalClassName⟩ ⟨ThesRelType⟩ ⟨LocalAttributeName⟩   ⟨LocalAttributeName⟩ ⟨ThesRelType⟩ ⟨LocalClassName⟩
⟨ThesRelType⟩	::=	<b>syn</b>   <b>bt</b>   <b>nt</b>   <b>rt</b>
⟨Module⟩	::=	<b>module</b> ⟨Identifier⟩ { ⟨OdlSpecification⟩ }