Se cambió la anterior definición

functionBody

: NATIVE\_ stringLiteral? SC

| ASYNC\_? EG expr SC

| ( ASYNC\_ ST? | SYNC\_ ST)? block

;

Por esta que se encuentra a continuación

functionBody

: ASYNC\_? EG expr SC

| ( ASYNC\_ ST? | SYNC\_ ST)? block

;

Se cambió la definición anterior

declaration

: ABSTRACT\_? (

EXTERNAL\_ factoryConstructorSignature

| EXTERNAL\_ constantConstructorSignature

| EXTERNAL\_ constructorSignature

| ( EXTERNAL\_ STATIC\_?)? getterSignature

| ( EXTERNAL\_ STATIC\_?)? setterSignature

| ( EXTERNAL\_ STATIC\_?)? functionSignature

| EXTERNAL\_? operatorSignature

| STATIC\_ CONST\_ type? staticFinalDeclarationList

| STATIC\_ FINAL\_ type? staticFinalDeclarationList

| STATIC\_ LATE\_ FINAL\_ type? initializedIdentifierList

| STATIC\_ LATE\_? varOrType initializedIdentifierList

| COVARIANT\_ LATE\_ FINAL\_ type? identifierList

| COVARIANT\_ LATE\_? varOrType initializedIdentifierList

| LATE\_? FINAL\_ type? initializedIdentifierList

| LATE\_? varOrType initializedIdentifierList

| redirectingFactoryConstructorSignature

| constantConstructorSignature ( redirection | initializers)?

| constructorSignature ( redirection | initializers)?

)

;

Por esta que se encuentra a continuación

declaration

: EXTERNAL\_ factoryConstructorSignature

| EXTERNAL\_ constantConstructorSignature

| EXTERNAL\_ constructorSignature

| ( EXTERNAL\_ STATIC\_?)? getterSignature

| ( EXTERNAL\_ STATIC\_?)? setterSignature

| ( EXTERNAL\_ STATIC\_?)? functionSignature

| EXTERNAL\_? operatorSignature

| STATIC\_ CONST\_ type? staticFinalDeclarationList

| STATIC\_ FINAL\_ type? staticFinalDeclarationList

| STATIC\_ LATE\_ FINAL\_ type? initializedIdentifierList

| STATIC\_ LATE\_? varOrType initializedIdentifierList

| COVARIANT\_ LATE\_ FINAL\_ type? identifierList

| COVARIANT\_ LATE\_? varOrType initializedIdentifierList

| LATE\_? FINAL\_ type? initializedIdentifierList

| LATE\_? varOrType initializedIdentifierList

| redirectingFactoryConstructorSignature

| constantConstructorSignature ( redirection | initializers)?

| constructorSignature ( redirection | initializers)?

;

Se cambio la definición anterior

initializerExpression

: conditionalExpression

| cascade

;

Por esta que se encuentra a continuación

initializerExpression

: assignableExpression assignmentOperator expr

| conditionalExpression

| cascade

| throwExpression

;

Se cambio la siguiente regla

identifier

: IDENTIFIER

| ABSTRACT\_

| AS\_

| COVARIANT\_

| DEFERRED\_

| DYNAMIC\_

| EXPORT\_

| EXTERNAL\_

| EXTENSION\_

| FACTORY\_

| FUNCTION\_

| GET\_

| IMPLEMENTS\_

| IMPORT\_

| INTERFACE\_

| LATE\_

| LIBRARY\_

| MIXIN\_

| OPERATOR\_

| PART\_

| REQUIRED\_

| SET\_

| STATIC\_

| TYPEDEF\_

| FUNCTION\_

| ASYNC\_

| HIDE\_

| OF\_

| ON\_

| SHOW\_

| SYNC\_

| AWAIT\_

| YIELD\_

| DYNAMIC\_

| NATIVE\_

;

Por la siguiente

identifier

: IDENTIFIER

| builtInIdentifier

| otherIdentifier

;

Se cambia esto

typeIdentifier

: IDENTIFIER

| ASYNC\_

| HIDE\_

| OF\_

| ON\_

| SHOW\_

| SYNC\_

| AWAIT\_

| YIELD\_

| DYNAMIC\_

| NATIVE\_

| FUNCTION\_

;

Por esto

typeIdentifier

: typeIdentifierNotType

| TYPEDEF\_

;

Se cambio la sgte. regla

libraryDeclaration

: libraryName? importOrExport\* partDirective\* (metadata topLevelDeclaration)\*

;

Por esta

libraryDeclaration

: scriptTag? libraryName? importOrExport\* partDirective\* (metadata topLevelDeclaration)\*

;

Se cambio esta regla

partDeclaration

: partHeader (metadata topLevelDeclaration)\*

;

Por esta

⟨*partDeclaration*⟩ ::= ⟨*partHeader*⟩ (⟨*metadata*⟩ ⟨*topLevelDeclaration*⟩)\* ⟨*EOF*⟩

Se cambio la sgte definición

typeNotVoid

: functionType QU?

| typeNotVoidNotFunction

;

Por esta

typeNotVoid

: functionType QU?

| typeNotVoidNotFunction QU?

;

Se cambio lo sgte

typeNotFunction

: VOID\_

| typeNotVoidNotFunction

;

Por esto

typeNotFunction

: VOID\_

| typeNotVoidNotFunction QU?

;

Se cambio esto

typeNotVoidNotFunction

: typeName typeArguments?

| FUNCTION\_ QU?

;

Por esto

typeNotVoidNotFunction

: typeName typeArguments?

| FUNCTION\_

;

Se agregaron las siguientes reglas de producción

typeIdentifierNotType

: IDENTIFIER

| otherIdentifierNotType

| DYNAMIC\_

;

builtInIdentifier

: ABSTRACT\_

| AS\_

| COVARIANT\_

| DEFERRED\_

| DYNAMIC\_

| EXPORT\_

| EXTERNAL\_

| EXTENSION\_

| FACTORY\_

| FUNCTION\_

| GET\_

| IMPLEMENTS\_

| IMPORT\_

| INTERFACE\_

| LATE\_

| LIBRARY\_

| MIXIN\_

| OPERATOR\_

| PART\_

| REQUIRED\_

| SET\_

| STATIC\_

| TYPEDEF\_

;

otherIdentifierNotType

: ASYNC\_

| HIDE\_

| OF\_

| ON\_

| SHOW\_

| SYNC\_

| AWAIT\_

| YIELD\_

;

otherIdentifier

: otherIdentifierNotType

| TYPEDEF\_

;

identifierNoDollar

: identifierStartNoDollar

| identifierPartNoDollar

;

reservedWord

: ASSERT\_

| BREAK\_

|CASE\_

|CATCH\_

|CLASS\_

|CONST\_

|CONTINUE\_

|DEFAULT\_

|DO\_

|ELSE\_

|ENUM\_

|EXTENDS\_

|FALSE\_

|FINAL\_

|FINALLY\_

|FOR\_

|IF\_

|IN\_

|IS\_

|NEW\_

|NULL\_

|RETURN\_

|SUPER\_

|SWITCH\_

|THIS\_

|THROW\_

|TRUE\_

|TRY\_

|VAR\_

|VOID\_

|WHILE\_

|WITH\_

;

singleLineComment

: ‘//‘ ~(lineBreak)\* (lineBreak)?

;

multiLineComment

: ‘/\*’ (multiLineComment

| ~ ‘\*/’)\* ‘\*/‘

;

singleLineString

: rawSingleLineString

| singleLineStringSQBeginEnd

| singleLineStringSQBeginMid expr (singleLineStringSQMidMid expr)\*

;

singleLineStringSQMidEnd

| singleLineStringDQBeginEnd

| singleLineStringDQBeginMid expr (singleLineStringDQMidMid expr)\*

;

rawSingleLineString

: 'r' ''' (~('''| '\r'|'\n'))\*'''

| 'r''"'(~('"'|'\r'|'\n'))\*'"'

;

stringContentCommon

: ~('\'|'''|'"'|'$'|'\r'|'\n')

| escapeSequence

| '\'~('n'|'r'|'b'|'t'|'v'|'x'|'u'|'\r'|'\n')

| simpleStringInterpolation

;

stringContentSQ

: stringContentCommon | ‘"'

;

singleLineStringSQBeginEnd

: ''' stringContentSQ\* '''

;

singleLineStringSQBeginMid

: ''' stringContentSQ\* '${'

;

singleLineStringSQMidMid

: '}' stringContentSQ\* '${'

;

singleLineStringSQMidEnd

: '}' stringContentSQ\* '''

;

stringContentDQ

: stringContentCommon

| '''

;

singleLineStringDQBeginEnd

: '"' stringContentDQ\* '"'

;

singleLineStringDQBeginMid

: '"' stringContentDQ\* '${'

;

singleLineStringDQMidMid

: '}' stringContentDQ\* '${'

;

singleLineStringDQMidEnd

: '}' stringContentDQ\* '"'

;

whitespace

: ('\t'|''|lineBreak)+

;