**宏定义**

**OPT\_CREATE\_ORIGINCLASS = 1**

**OPT\_CREATE\_SELECTDEPUTY = 2**

**OPT\_DROP = 3**

**OPT\_INSERT = 4**

**OPT\_DELETE = 5**

**OPT\_SELECT\_DERECTSELECT = 6**

**OPT\_SELECT\_INDERECTSELECT = 7**

**词法分析**

**类名 属性名以小写字母开头，包含字母和数字**

**string类型不包含空格**

<SEMICOLON:";">

|<CREATE:"CREATE">

|<DROP:"DROP">

|<CLASS:"CLASS">

|<INSERT:"INSERT">

|<INTO:"INTO">

|<VALUES:"VALUES">

|<LEFT\_BRACKET:"(">

|<COMMA:",">

|<RIGHT\_BRACKET:")">

|<DELETE:"DELETE">

|<FROM:"FROM">

|<WHERE:"WHERE">

|<SELECT:"SELECT">

|<SELECTDEPUTY:"SELECTDEPUTY">

|<ID: ["a"-"z"](["a"-"z","0"-"9"])\* >

|<EQUAL:"=">

|<INT: "0"|(["1"-"9"](["0"-"9"])\*) >

|<STRING: "\""(["a"-"z","A"-"Z","1"-"9"])\*"\"" >

|<CROSS:"->">

|<DOT:".">

|<PLUS:"+">

|<AS:"AS">

**语法分析**

编译SQL的产生式

大写为终结符，小写为非终结符

1. sql -> create

| drop

| insert

| delete

| select

1. create-> CREATE originclass SEMICOLON

| CREATE selectdeputy SEMICOLON

1. drop->DROP CLASS classname SEMICOLON
2. insert->INSERT INTO classname VALUES LEFT\_BRACKET value (COMMA value)\* RIGHT\_BRACKET SEMICOLON
3. delete->DELETE FROM classname WHERE condition SEMICOLON
4. select->SELECT directselect SEMICOLON

| SELECT indirectselect SEMICOLON

1. originclass-> CLASS classname LEFT\_BRACKET attr attrtype (COMMA attr attrtype)\* RIGHT\_BRACKET
2. selectdeputy-> SELECTDEPUTY classname SELECT directselect
3. classname->ID
4. attrtype->ID
5. attr->ID
6. condition->attr EQUAL INT

| attr EQUAL STRING

1. value->INT

| STRING

1. directselect->attr (COMMA attr)\* FROM classname WHERE condition
2. indirectselect-> classname1 (CROSS classname)\* DOT attr FROM classname1 WHERE condition

**语义制导翻译**

1. sql -> create

sql.opt\_result =create.opt\_type+ create.attr\_count+ create.classname+create.attr+( create.isSwitch+create.switchValue +create.dAttr)+（create.selectclassname+ create.condition）

1. sql ->drop

sql.opt\_result = drop.opt\_type+drop.classname

1. sql -> insert

sql.opt\_result = insert.opt\_type+ insert.classname+ insert.value\_count+ insert.value

1. sql -> delete

sql.opt\_result = delete.opt\_type+ delete.classname+ delete.condition

1. sql -> select

sql.opt\_result =select.type+select.classname+select.condition+(select.attr\_count)+

select.attr+(select.isSwitch+select.switchValue+select.dAttr)+(select.class\_count+select.crossclass)

1. create-> CREATE originclass SEMICOLON

create.classname = originclass.classname

attr\_count = attr.count

attr = attr

create.opt\_type = OPT\_CREATE\_ORIGINCLASS

1. create-> CREATE selectdeputy SEMICOLON

create.classname = selectdeputy.name

create.attr = selectdeputy.attr

create.attr\_count = selectdeputy.attr\_count

create.condition = selectdeputy.condition

create.selectclassname = selectdeputy.selectclassname

create.opt\_type = OPT\_CREATE\_SELECTDEPUTY

1. drop->DROP CLASS classname SEMICOLON

drop.classname = classname.name

drop.opt\_type = OPT\_DROP

1. insert->INSERT INTO classname VALUES LEFT\_BRACKET value (COMMA value)\* RIGHT\_BRACKET SEMICOLON

insert.classname = classname.lexvar

insert.value = value.name+REPEAT(\*)

insert.value\_count = COUNT(\*)

insert.opt\_type = OPT\_INSERT

1. delete->DELETE FROM classname WHERE condition SEMICOLON

delete.classname = classname.name

delete.condition = condition.name

delete.type = OPT\_DELETE

1. select->SELECT directselect SEMICOLON

select.attr = directselect. attr

select.isSwitch = directselect.isSwitch

select.switchValue = directselect.switchValue

select.dAttr = directselect.dAttr

select.attr\_count = directselect.attr\_count

select.classname = directselect.classname

select.condition = directselect.condition

select.type = OPT\_SELECT\_DIRECTSELECT

1. select-> SELECT indirectselect SEMICOLON

select. attr = indirectselect. attr

select.class\_count = indirectselect.class\_count

select.crossclass = indirectselect.crossclass

select.classname = indirectselect.classname

select.condition = indirectselect.condition

select.type = OPT\_SELECT\_INDIRECTSELECT

1. originclass-> CLASS classname LEFT\_BRACKET attr attrtype (COMMA attr attrtype)\* RIGHT\_BRACKET

originclass.classname = classname.name

originclass.attr\_count = 1+COUNT(\*)

originclass.attr = attr.name +attrtype.name+REPEAT(\*)

1. selectdeputy-> SELECTDEPUTY classname SELECT directselect

selectdeputy.classname = classname.name

selectdeputy.attr\_count = directselect.attr\_count

selectdeputy.isSwitch = directselect.isSwitch

selectdeputy.switchValue = directselect.switchValue

selectdeputy.attr = directselect.attr

selectdeputy.dAttr = directselect.dAttr

selectdeputy.condition = directselect.condition

selectdeputy.selectclassname= directselect.classname

1. classname->ID

classname.name = ID.lexval

1. attrtype->ID

attrtype.name = ID.lexval

1. attr->ID

attrtype.name = ID.lexval

1. dAttr->ID

attrtype.name = ID.lexval

1. condition->attr EQUAL INT

condition.name = attr.name+”,”+EQUAL.lexval+”,”+INT.lexval

1. condition-> attr EQUAL STRING

condition.name = attr.name+”,”+EQUAL.lexval+”,”+STRING.lexval

1. value->INT

value.name =INT.lexval

1. value->STRING

value.name =STRING.lexval

1. directselect->attr [PLUS value] AS dAttr (COMMA attr [PLUS value] AS dAttr)\* FROM classname WHERE condition

directselect. attr = attr.name +REPEAT(\*)

directselect.isSwitch = 如果有[]里的，置为1，否则为0 +REPEAT(\*)

directselect.switchValue = value.name

directselect.dAttr = dAttr.name+REPEAT（\*）

directselect.attr\_count = 1+COUNT(\*)

directselect.classname = classname.name

directselect.condition = condition.name

directselect.isSwitch = 如果有[]里的，置为1，否则为0

1. indirectselect-> classname1 (CROSS classname)\* DOT attr FROM classname1 WHERE condition

indirectselect. attr = attr.name

indirectselect.class\_count = COUNT(\*)

indirectselect.crossclass = classname.name+REPEAT(\*)

indirectselect.classname = classname1.name

indirectselect.condition = condition.name