Follow Sets for the ERPLAG Language

<program>	{$}
<moduleDeclarations>	{DEF, DRIVERDEF}
<moduleDeclaration>	{DECLARE, DEF, DRIVERDEF}
<otherModules>	{DRIVERDEF, $}
<driverModule>	{DEF, $}
<module>	{DEF, DRIVERDEF, $}
<ret>	{START}
<input_plist>	{SQBC}
<sub_input_plist>	{SQBC}
<output_plist>	{SQBC}
<sub_output_plist>	{SQBC}
<dataType>	{SEMICOL, COMMA, SQBC}
<dynamic_range>	{SQBC}
<type>	{SEMICOL, COMMA, SQBC}
<moduleDef>	{DEF, DRIVERDEF, $}
<statements>	{END, BREAK}
<statement>	{GET_VALUE, PRINT, ID, SQBO, USE, DECLARE, SWITCH, FOR, WHILE, END, BREAK}
<ioStmt>	{GET_VALUE, PRINT, ID, SQBO, USE, DECLARE, SWITCH, FOR, WHILE, END, BREAK}
<extended_var>	{BC}
<var>	{MUL, DIV, SEMICOL, MUL, DIV, PLUS, MINUS, SEMICOL, BC, AND, OR, LT, LE, GT, GE, EQ, NE}

<whichId>	{MUL, DIV, SEMICOL, MUL, DIV, PLUS, MINUS, SEMICOL, BC, AND, OR, LT, LE, GT, GE, EQ, NE}
<simpleStmt>	{GET_VALUE, PRINT, ID, SQBO, USE, DECLARE, SWITCH, FOR, WHILE, END, BREAK}
<assignmentStmt>	{GET_VALUE, PRINT, ID, SQBO, USE, DECLARE, SWITCH, FOR, WHILE, END, BREAK}
<whichStmt>	{GET_VALUE, PRINT, ID, SQBO, USE, DECLARE, SWITCH, FOR, WHILE, END, BREAK}
<lvalueIDStmt>	{GET_VALUE, PRINT, ID, SQBO, USE, DECLARE, SWITCH, FOR, WHILE, END, BREAK}
<lvalueARRStmt>	{GET_VALUE, PRINT, ID, SQBO, USE, DECLARE, SWITCH, FOR, WHILE, END, BREAK}
<index>	{SQBC, RANGEOP}
<moduleReuseStmt>	{GET_VALUE, PRINT, ID, SQBO, USE, DECLARE, SWITCH, FOR, WHILE, END, BREAK}
<optional>	{USE}
<idList>	{SQBC, SEMICOL, COLON}
<sub_idList>	{SQBC, SEMICOL, COLON}
<expression>	{SEMICOL}
<arithmeticExpr>	{SEMICOL, BC, AND, OR, LT, LE, GT, GE, EQ, NE}
<sub_arithmeticExpr>	{BO, ID, NUM, RNUM}
<term>	{PLUS, MINUS, SEMICOL, BC, AND, OR, LT, LE, GT, GE, EQ, NE}
<sub_term>	{BO, ID, NUM, RNUM, MUL, DIV, PLUS, MINUS, SEMICOL, BC, AND, OR, LT, LE, GT, GE, EQ, NE}

<factor>	{MUL, DIV, PLUS, MINUS, SEMICOL, BC, AND, OR, LT, LE, GT, GE, EQ, NE}
<sub_factor>	{MUL, DIV, PLUS, MINUS, SEMICOL, BC, AND, OR, LT, LE, GT, GE, EQ, NE}
<op1>	{BO, ID, NUM, RNUM}
<op2>	{BO, ID, NUM, RNUM, MUL, DIV, PLUS, MINUS, SEMICOL, BC, AND, OR, LT, LE, GT, GE, EQ, NE}
<booleanExpr>	{SEMICOL, BC}
<sub_booleanExpr>	{SEMICOL, BC, AND, OR}
<logicalOp>	{BO, ID, NUM, RNUM}
<relationalOp>	{BO, ID, NUM, RNUM}
<declareStmt>	{GET_VALUE, PRINT, ID, SQBO, USE, DECLARE, SWITCH, FOR, WHILE, END, BREAK}
<conditionalStmt>	{GET_VALUE, PRINT, ID, SQBO, USE, DECLARE, SWITCH, FOR, WHILE, END, BREAK}
<caseStmt>	{DEFAULT, END}
<nullableCaseStmt>	{DEFAULT, END}
<value>	{COLON}
<default>	{END}
<iterativeStmt>	{GET_VALUE, PRINT, ID, SQBO, USE, DECLARE, SWITCH, FOR, WHILE, END, BREAK}
<range>	{BC}