FIRST SETS

FIRST(<CONST>) = { INT\_CONST, FLOAT\_CONST , STRING\_CONST , CHAR\_CONST ,BOOL\_CONST }

FIRST(<ID\_Const>) = { ID, INT\_CONST , FLOAT\_CONST , STRING\_CONST , CHAR\_CONST , BOOL\_CONST }

FIRST(<Static>) = {Static}

FIRST(<Access\_Modifier>) = { access\_modifier, Null}

FIRST(<Return\_Type>) = { void, DT }

FIRST(<M\_ST>) = { jabtak , DT , Barbar , agar , return , inc\_dec , ID , break , continue, this , Null}

FIRST(<Body>) = {; , { , jabtak , DT , Barbar , agar , return , inc\_dec , ID , break , continue , this }

FIRST(<List\_Param>) = {DT , Null}

FIRST(<List\_Param1>) = {, , Null}

//FIRST(<Param>) = { ID, INT\_CONST , FLOAT\_CONST , STRING\_CONST , CHAR\_CONST , BOOL\_CONST , ! , ( , inc\_dec , Null}

FIRST(<Param1>) = {, , Null}

FIRST(<Method\_Call\_1>) = { ( }

**Single Statement:**

FIRST(S\_ST) = {jabtak , DT , Barbar , agar , return , inc\_dec , ID , break , continue , this}

//FIRST(<S\_St\_ID>) = {inc\_dec , AOP , ID , [ , . , ( , = }

FIRST(<Array\_Access >) = { ID, INT\_CONST , FLOAT\_CONST , STRING\_CONST , CHAR\_CONST , BOOL\_CONST , ! , ( , inc\_dec , ] }

FIRST(<S\_St\_DT>) = {ID , void , DT , [}

//FIRST(<S\_St\_DT2>) = { AOP , , , ; }

**Variable Declaration:**

FIRST(<DEC>) = { DT}

FIRST(<Variable\_Link>) = {ID}

FIRST(<Variable\_Link2> ) = {AOP , , ;}

//FIRST(<Variable\_Value>) = { ID, INT\_CONST , FLOAT\_CONST , STRING\_CONST , CHAR\_CONST , BOOL\_CONST , ! , ( , inc\_dec }

FIRST(<LIST >) = {, , ;}

**Assignment of Variable:**

//FIRST(<Assign\_Op>) = { AOP }

//FIRST(<Assign\_Op2>) = { ID, INT\_CONST , FLOAT\_CONST , STRING\_CONST , CHAR\_CONST , BOOL\_CONST , ! , ( , inc\_dec }

**If Else Condition:**

FIRST(<agar\_warna>) = {agar}

FIRST(<O\_Else>) = {warna , Null}

**While Statement:**

FIRST(<Jab\_tak>) = {jabtak}

**Return:**

FIRST(<Return>) = {return}

FIRST(<Return2>) = { ; , ID, INT\_CONST , FLOAT\_CONST , STRING\_CONST , CHAR\_CONST , BOOL\_CONST , ! , ( , inc\_dec }

**Break:**

FIRST(<Break>) = {break}

**Continue:**

FIRST(<Continue>) = {continue}

**This:**

FIRST(<This>) = {this}

FIRST(<LIST1>) = {; , AOP}

FIRST(<LIST2>) = { ID, INT\_CONST , FLOAT\_CONST , STRING\_CONST , CHAR\_CONST , BOOL\_CONST }

**Class Declaration:**

FIRST(<Class\_Dec>) = { access\_modifier, class}

FIRST(<Class\_Link>) = {class}

FIRST(<Class\_Base>) = {Null , :}

FIRST(<Class\_Body>) = { access\_modifier , static , DT ,void ,ID , class , Null}

FIRST(<Class\_ Member >) = { access\_modifier , static , DT ,void ,ID , class }

FIRST(<Member\_Link>) = { static , DT ,void ,ID , class }

//FIRST(<Object\_Constructor\_DEC>) = { ID, [ , (}

FIRST(<DT\_2>) = {ID , [}

//FIRST(<ID\_1>) = {( , AOP , , , ; }

FIRST(<SS\_A>) = {DT , ID , void}

FIRST(<DT\_1>) = {ID , [}

//FIRST(<ID\_2>) = {( , AOP , , , ; }

FIRST(<Id\_OArray>) = {ID , [}

FIRST(<DT\_A>) = { AOP , , , ; , (}

//FIRST(<Id\_A>) = {( , = , ;}

**Constructor Declaration:**

//FIRST(<Constructor\_DEC>) = { ( }

**Array Declaration:**

FIRST(<Array\_DEC>) = {[}

FIRST(<INIT\_Array>) = {; , =}

FIRST(<Array\_const>) = {{ , ;}

FIRST(<Array\_C>) = { { }

FIRST(<Array\_C2>) = {, , } }

**Method Declaration:**

FIRST(<Method\_DEC>) = {DT , void}

FIRST(<Method\_Link 3>) = { ( }

**Object Declaration**

//FIRST(<Object\_Link>) = {ID , [}

//FIRST(<object\_array\_dec>) = { = }

**//FIRST(<obj\_arr\_dec1>) = { ; , { }**

**//FIRST(<obj\_arr\_dec2>) = { new , Null}**

**//FIRST(<obj\_arr\_dec3>) = { , , }}**

//FIRST(<Object\_Creation\_Exp>) = {= , ;}

//FIRST(<Object\_List>) = { , }

**Object Calling:**

FIRST(<Object\_Call>) = {. , [}

**FOR LOOP:**

FIRST(<Bar\_Bar>) = {barbar}

FIRST(<F1>) = {DT , ID , Null}

//FIRST(<F2>) = { ID, INT\_CONST , FLOAT\_CONST , STRING\_CONST , CHAR\_CONST , BOOL\_CONST , ! , ( , inc\_dec, Null }

FIRST(<X>) = { , , Null}

FIRST(<F3>) = {inc\_dec , ID , Null}

FIRST(<F4>) = {inc\_dec , AOP}

**Expression:**

FIRST(<Exp>) = { ID, INT\_CONST , FLOAT\_CONST , STRING\_CONST , CHAR\_CONST , BOOL\_CONST , ! , ( , inc\_dec }

FIRST(<OR\_Exp>) = { ID, INT\_CONST , FLOAT\_CONST , STRING\_CONST , CHAR\_CONST , BOOL\_CONST , ! , ( , inc\_dec }

FIRST(<OR\_Exp2>) = {|| , Null}

FIRST(<AND\_Exp>) = { ID, INT\_CONST , FLOAT\_CONST , STRING\_CONST , CHAR\_CONST , BOOL\_CONST , ! , ( , inc\_dec }

FIRST(<AND\_Exp2>) = {&& , Null}

FIRST(<ROP>) = { ID, INT\_CONST , FLOAT\_CONST , STRING\_CONST , CHAR\_CONST , BOOL\_CONST , ! , ( , inc\_dec }

FIRST(<ROP2>) = {ROP , Null}

FIRST(<E>) = { ID, INT\_CONST , FLOAT\_CONST , STRING\_CONST , CHAR\_CONST , BOOL\_CONST , ! , ( , inc\_dec }

FIRST(<E2 >) = {Plus\_Minus , Null}

FIRST(<T>) = { ID, INT\_CONST , FLOAT\_CONST , STRING\_CONST , CHAR\_CONST , BOOL\_CONST , ! , ( , inc\_dec }

FIRST(<T2>) = { M\_D\_M , Null}

//FIRST(<F>) = { ID, INT\_CONST , FLOAT\_CONST , STRING\_CONST , CHAR\_CONST , BOOL\_CONST , ! , ( , inc\_dec }

FIRST(<inc\_dec\_list>) = { [ , . , Null}

FIRST(<id\_op>) = { Null , ( , [ , . , inc\_dec}

FIRST(<Member\_exp>) = { . }

FIRST(< Member\_exp\_2>) = {Null , ( , [}