<class\_def> → <class\_header> class ID <extend\_st> <implement\_st> : ( <class\_body> )

**Agar AM kch bhi nhi h to bydefault public hoga**

<class\_header> 🡪 AM <ch2> | final | €

<ch2> 🡪 final | €

<extend\_st> → extends ID |  €

<implement\_st> → implements ID <interface\_rec> |  €

<interface\_rec> → , ID <interface\_rec> |  €

<class\_body> → AM <amh2> <cb2>

<class\_body> → static <amh3> <cb2>

<class\_body> → final <cb2>

<class\_body> → <cb2>

<class\_body> → AM ID { <params\_list> } : ( <constr\_body> ) <class\_body>

<class\_body> → €

**final**

<class\_body> → AM <cb3>

<class\_body> → static <amh3> <cb2>

<class\_body> → final <cb2>

<class\_body> → <cb2>

<class\_body> → €

<cb2> 🡪 <dec> <class\_body> | <func\_def> <class\_body> hogai

<cb3> 🡪 <amh2> <cb2> | ID { <params\_list> } : ( <constr\_body> ) <class\_body> hogai

Yaha aayie

<cb2> 🡪 DT <cb2a>

<cb2> 🡪 String <cb2a>

<cb2> 🡪 dict <cb2b>

<cb2> 🡪 ID <cb2c>

<cb2> 🡪 void ID { <params\_list> } : <body> <class\_body>

<cb2a> 🡪 ID <dec1> <class\_body> | <fnd2> <class\_body> hogai neche

<cb2b> 🡪 ID <init\_dict> <list\_dict> <class\_body> | <fnd2> <class\_body> hogai neche

<cb2c> 🡪 ID <init> <list> <class\_body> | <fnd2> <class\_body> hogai neche

Ye hogai

<cb2a> 🡪 ID <cb2a1>

<cb2a> 🡪 [ ] <arr\_mul> ID { <params\_list> } : <body> <class\_body>

<cb2a1> 🡪 <dec1> <class\_body> | { <params\_list> } : <body> <class\_body>

Ye hogai

<cb2b> 🡪 ID <cb2b1>

<cb2b> 🡪 [ ] <arr\_mul> ID { <params\_list> } : <body> <class\_body>

<cb2b1> 🡪 <init\_dict> <list\_dict> <class\_body> | { <params\_list> } : <body> <class\_body>

Ye hogai

<cb2c> 🡪 ID <cb2c1>

<cb2c> 🡪 [ ] <arr\_mul> ID { <params\_list> } : <body> <class\_body>

<cb2c1> 🡪 <init> <list> <class\_body> | { <params\_list> } : <body> <class\_body>

**Ye hogai**

<cb3> 🡪 DT <cb2a>

<cb3> 🡪 String <cb2a>

<cb3> 🡪 dict <cb2b>

<cb3> 🡪 ID <cb3a>

<cb3> 🡪 void ID { <params\_list> } : <body> <class\_body>

<cb3> 🡪 static <amh2a> <cb2>

<cb3> 🡪 final <cb2>

<cb3a> 🡪 <cb2c> | { <params\_list> } : ( <constr\_body> ) <class\_body>

<dec> 🡪 **DT ID <init\_var> <list\_var>**

<dec> 🡪 **String ID <init\_var> <list\_var>**

<dec> 🡪 **DT ID [ <OE> ] <arr\_size> <init \_arr> <list\_arr>**

<dec> 🡪 **String ID[ <OE> ]<arr\_size> <init\_arr> <list\_arr>**

**<dec> 🡪 DT ID <dec1> | String ID <dec1>**

**<dec1> 🡪 <init\_var> <list\_var> | [ <OE> ] <arr\_size> <init \_arr> <list\_arr>**

<dec> 🡪 **ID ID <init\_obj> <list\_obj>**

<dec> 🡪 **ID ID <option> <init\_enum> <list\_enum>**

<dec> 🡪 <dec\_dict>

<dec> 🡪 ID ID <dec2>

<dec2> 🡪 **<list\_obj>**

<dec2> 🡪 **= new ID { <args-list> } <list\_obj>**

**<dec2> 🡪 <list\_enum>**

**<dec2> 🡪 = <OE> <list\_enum>**

<attrb&method\_header> → AM <amh2> | static <amh3> | final | €

<amh2> 🡪 € | static <amh2a> | final

<amh2a> 🡪 final | €

<amh3> 🡪 final | €

<attrb> → <attrb&method\_header> <dec>

<method> → <attrb&method\_header> <func\_def>

<constr> → AM ID { <params\_list> } : ( <constr\_body> )

<constr\_body> →  <pointer\_st> <MST>

<pointer\_st> → TS { <args\_list> } <pointer\_st> |  €

<args\_list> 🡪 <OE> <list\_args> | ∈

<list\_args> 🡪 , <OE> <list\_args> | ∈