SLR parser is a type of LR parser with small parse table and a relatively small parser generator ~~algo~~ SLR ~~generation~~ generator accept fewer grammer than do LALR generators like yacc and Bison

Construction of SLR parsing table

1] Construct $C = \{ I_0, I_1, \& \ldots I_n \}$, the collections of sets of LR(0) items for C'.

2] ~~If~~ State i is constructed from I_i. The parsing action for state i are determined as follows

→ If $[A \to ?.a?]$ is in I_i and GOTO $(I_i, a) = I_j$, then set ACTION $[i, a]$ to "shift j". Here a must be terminal~~s~~

→ If $[A \to ?]$ is in I_j, then set ACTION $[i, a]$ to "reduce $A \to ?$" for all a in FOLLOW (A); here A may not be S'.

→ If $[S \to S]$ is in I_i, then set action $[i, \$]$ to "accept"

If any conflicting actions are generated by the above rules we say that grammer is not SLR

3] The goto-rule transition for state i are constructed for all non-terminal A using the rules.
if GTO $(I_i, A) = I_j$ then GOTO $[i, A] = j$

4] All entries not defined by rules 2 & 3 are made error