Consider the following grammar. (number) represent production umber of grammar.

S-> NP VP (1)
S-> X NP VP (2)
S-> VP (3)
NP-> D N (4)
NP-> V (5)
VP-> V NP (6)
D-> this (7)
D-> that (8)
D-> a (9)
D-> the (10)
N-> book (11)
N-> flight (12)
N-> john (13)
N-> ball (14)
N-> meal (15)
V-> book (16)
V-> include (17)
V-> read (18)
X-> does (19)
X->a (20)

Its SLR-1 table is given below. (more clear image can be found in folder)

	$	a	does	read	include	book	flight	meal	ball	john	the	that	this	S	VP	NP	D	N	V	X
0		s14	s13	s12	s11	s10					s9	s8	s7	s6	s5	s4	s3		s2	s1
1		s24									s9	s8	s7			s23	s3			
2	r(VP → V)	s24									s9	s8	s7			s22	s3			
3						s21	s20	s19	s18	s17								s16		
4				s12	s11	s10									s15				s2	
5	r(S → VP)																			
6	acc																			
7						r(D → this)	r(D → this)	r(D → this)	r(D → this)	r(D → this)										
8						r(D → that)	r(D → that)	r(D → that)	r(D → that)	r(D → that)										
9						r(D → the)	r(D → the)	r(D → the)	r(D → the)	r(D → the)										
10	r(V → book)	r(V → book)									r(V → book)	r(V → book)	r(V → book)							
11	r(V → include)	r(V → include)									r(V → include)	r(V → include)	r(V → include)							
12	r(V → read)	r(V → read)									r(V → read)	r(V → read)	r(V → read)							
13		r(X → does)									r(X → does)	r(X → does)	r(X → does)							
14		r(X → a)				r(D → a)	r(D → a)	r(D → a)	r(D → a)	r(D → a)	r(X → a)	r(X → a)	r(X → a)							
15	r(S → NP VP)																			
16	r(NP → D N)			r(NP → D N)	r(NP → D N)	r(NP → D N)														
17	r(N → john)			r(N → john)	r(N → john)	r(N → john)														
18	r(N → ball)			r(N → ball)	r(N → ball)	r(N → ball)														
19	r(N → meal)			r(N → meal)	r(N → meal)	r(N → meal)														
20	r(N → flight)			r(N → flight)	r(N → flight)	r(N → flight)														
21	r(N → book)			r(N → book)	r(N → book)	r(N → book)														
22	r(VP → V NP)																			
23				s12	s11	s10									s25				s2	
24						r(D → a)	r(D → a)	r(D → a)	r(D → a)	r(D → a)										
25	r(S → X NP VP)																			

Parsing table can be found in table.csv file,where columns are separated by commas (,) .
You need to implement SLR-1 parsing algorithm for this parsing table, consider input already tokenized (no lexical analysis required).

Each stack operation should be properly displayed.