⇒ How to check that Given Grammar is LL(1) or not:

using the parsing Table.

	First()	follow()
P1 P2 S → (L) | a	(a	{$,) }
P3 P4 L → SL'	(a	{) }
P5 L' → ε / , SL	ε ,	{) }

→ Note from Given grammar first we have to find the first() and follow of the given grammar.

→ first() is easy to find lets move towards follow.

→ for follow() start from S and then check where S is present in the whole grammar? it is present in the second line and S follow L' and we will write the L' first which is ε and) so as we dont write ε first we write {$} dollar sign and then we will write {)} and then due to ε we will remove the S & ε and the we will write its source follow which is L and L follow).

→ L check where the L is present in the whole grammar, it is present in the first line and it follow {)} so we will write L → SL'

→ Now for L' and check where the L' is present in the whole grammar, Yes it is

present in the second line. and it follow nothing so we will write the its source L and it follows {) $ }

→ Now let's create the Parse Table.

		(	)	a	,	$
S		1		2		
L		~3		~3		
L'		~	4		5	

→ Step1 On the left hand side we write all variable which is present in the CFG.

→ Step2 write all the terminals in the row (record) Note { $ } dollar is always terminal.

→ Step3 Now we have to filled all the productions P_1, P_2, P_3, P_4 and P_5 are the productions.

How we will fill ??

→ our first production is S→(L) S goes to L and (L) is in the right hand side, so we will write first of it. which is { (} and write on that column as 1.

→ Now an-other production which is S→a and it's first is a.

→ Now for production three. which is L and L goes to SL' means L→ SL' and we have to find the right hand side first

which is {C} and {a} and it is production no 3 so write 3 and 3 and those columns.

→ Now for production no 4 and 5 and see $L' \to \varepsilon / SL'$ note 4 production rule is $L' \to \varepsilon$ L' goe to ε so when ε is coming it is an exceptional case.
beause first of the ε is always ε and ε is not included in the table so whenever ε is coming, in this case we have to write it's source follow()

which is L' and it's follow is {)} and write 4 there.
tho is how we have to fill.

→ Now production no 5 which is L' goes to SL' and its first is S and S first is {a} coming

This is how we made the passe table.

→ Note:- in every box there is only one Entry come so it is known as LL(1) Grammar.

→ ex2: from the Give grammar check weather the Given grammar is LL(1) or Not:

	First()	Follow()
S → aSbS / bSaS / ε	a, b, ε	{$, a, $}

step1: first we will find the follow() and First():
for follow check S where is present tes it is present in first line and it follow is b and in second production. S follow to a and in third ε instead we write ε for it

Now lets create the passing Table.
we have one variable with three production rule

	a	b	$
S	1	2	3
	3	3	

So in this case one column has more Entries occurs so it is consider as not a LL(1). grammar.