a)

Consider the data:

$$CNF_k = \left\{ \begin{array}{l} \langle\phi\rangle \mid \phi \text{ is a satisfiable } cnf - \text{formula where each variable} \\ \text{appears in at most } k \text{ places} \end{array} \right\}$$

CNF is Conjunctive normal form; it contains few rules.

- A literal is Boolean variable or negated Boolean variable in the form
- Clause contains several literals connected with $\vee$s and $\wedge$s.

Now have to show that $CNF_2 \in P$.

Class-P: P is a class of languages that are decidable in polynomial time on a deterministic single $-$tape Turing $-$machine.

Let T_p be the polynomial time decider for CNF_2.

T_p can be described as follows:

$T_p =$ "on input $\langle\phi\rangle$":

According to CNF rules, choose the clauses:

1. Consider the first clause of ϕ. If it is of the form x, and there is a clause $\neg x$ in ϕ, reject.
2. CNF is the form $x \vee A$, where A is CNF. If x does not appear negated in other clauses, remove every clause of the form $x \vee B$ of ϕ and calculate the result ϕ, if there is no clauses in ϕ then accept.
3. Solve CNF where c occurs in every clause, where negation of c does not appear.
4. When searching with ϕ, if clauses found in the form $x \vee A$ and $\sim x \vee B$ then remove. Add $A \vee B$ in ϕ
5. Go to step 1.

Every time T_p processes each variable and reaches either accept or reject. Because of this the number of clauses in ϕ might decrease by 1 or 2. Hence running time of T_p becomes polynomial time in terms of the number of variables.

So, $\boxed{CNF_2 \in P}$.

b)

Now have to show that CNF_3 is NP-complete.

$NP-$complete: A language B is $NP-$complete if is satisfies two conditions:

1. B is in NP
2. Every A in NP is polynomial time reducible to B.

Step 1: $CNF_3 \in NP$: If CNF_3 is in NP

- V_p is a verified in polynomial time and it is described as follows:
  - $V_p =$ "on input $\langle\langle\phi\rangle, x\rangle$"

According to CNF rules, verify the following clauses:

- Verify each variable in ϕ which occurs in at most 3 places.
- Verify whether x is a satisfying assignment in ϕ.
- If both conditions are satisfied, then accept.
- Otherwise, reject.

Step 2: $\underline{3SAT \le_p CNF_3}$: It is best example for CNF_3 satisfying assignment.

Let r_p be the polynomial time reduction from $3SAT$ to CNF_3.

When an input instance ϕ of $3SAT, r_p(\langle\phi\rangle)$ is given then construct an instance of CNF_3 from the following:

- First read from left to right, select the best example variable that access more than three times in the formula. Example variable as S occurs in m multiple places. $(x_1 \vee A_1),\ldots(x_m \vee A_m)$ where x_i is S or negated S.
- If nothing results more than three times, then output is ϕ
- Select variables $S_1,\ldots,S_m$. If any $(x_i \vee A_i)$ remove from the formula
- $(S_1 \vee A_1) \wedge (\neg S_1 \vee S_2) \wedge (S_2 \vee A_2) \wedge (\neg S_2 \vee S_3) \ldots (S_m \vee A_m) \wedge (\neg S_m \vee S_m)$
- Go to step 1

Obviously reduced polynomial time $r_p(\langle\phi\rangle)$ is a formula identified that every variable occurs at most three times. It is also clear that ϕ is satisfiable if and only if $r_p(\langle\phi\rangle)$ is satisfiable. The r_p is a reduced polynomial time in terms of the number of variable in ϕ from (1) and (2) $\boxed{CNF_3 \text{ is } NP\text{-complete}}$.