AI Tutorial 6

- Write a program to solve N-Queens problem using Prolog.

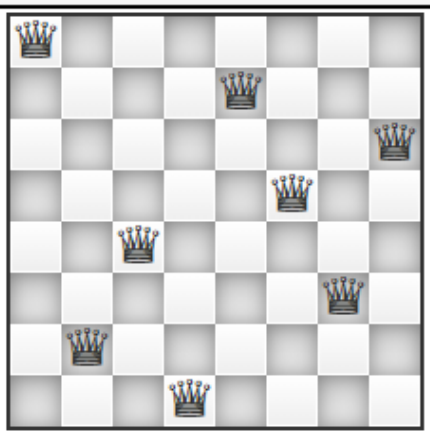
Code:-

```
:- use_rendering(chess).

queens(N, Queens) :-
    length(Queens, N),
      board(Queens, Board, 0, N, _, _),
      queens(Board, 0, Queens).

board([], [], N, N, _, _).
board([_|Queens], [Col-Vars|Board], Col0, N, [_|VR], VC) :-
      Col is Col0+1,
      functor(Vars, f, N),
      constraints(N, Vars, VR, VC),
      board(Queens, Board, Col, N, VR, [_|VC]).

constraints(0, _, _, _) :- !.
constraints(N, Row, [R|Rs], [C|Cs]) :-
      arg(N, Row, R-C),
      M is N-1,
      constraints(M, Row, Rs, Cs).

queens([], _, []).
queens([C|Cs], Row0, [Col|Solution]) :-
      Row is Row0+1,
      select(Col-Vars, [C|Cs], Board),
      arg(Row, Vars, Row-Row),
      queens(Board, Row, Solution).
```

Output :-

queens(8, Queens).

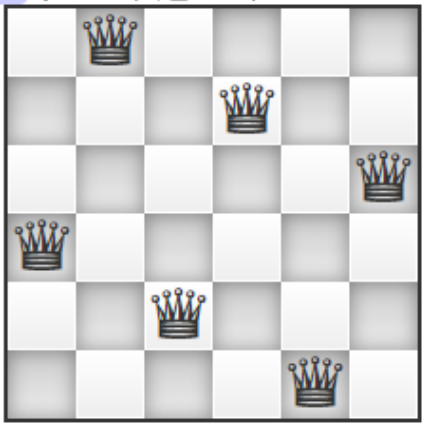
Queens =



trace, (*queens*(6, Queens)).

Call: *queens*(6, _8896)

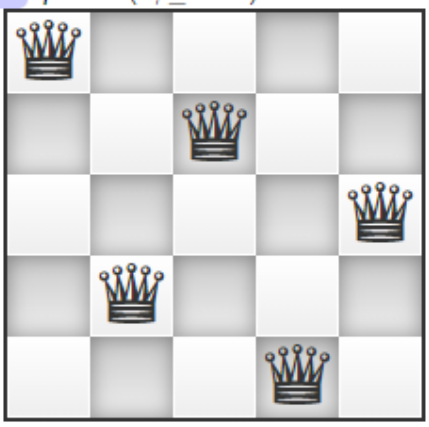
Queens =

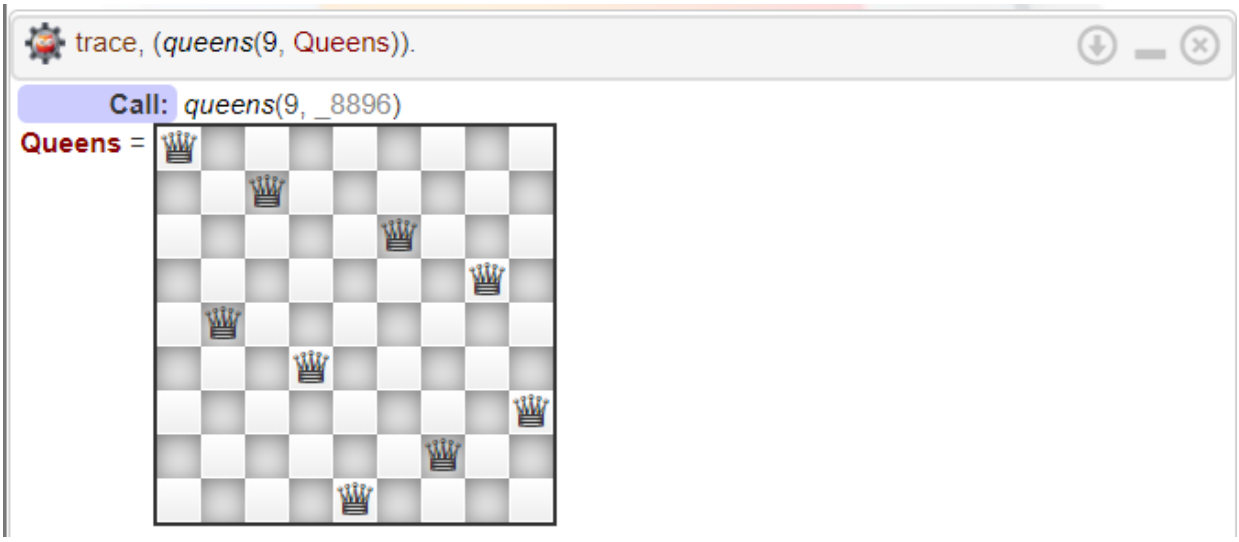


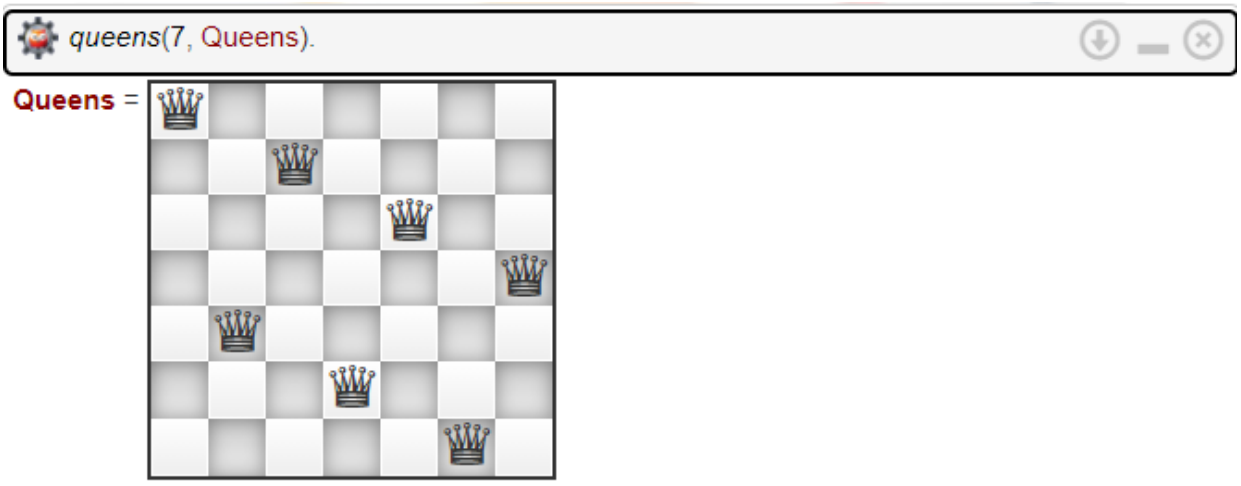
trace, (*queens*(5, Queens)).

Call: *queens*(5, _8896)

Queens =




trace, (queens(9, Queens)).
Call: queens(9, _8896)
Queens =


queens(7, Queens).
Queens =