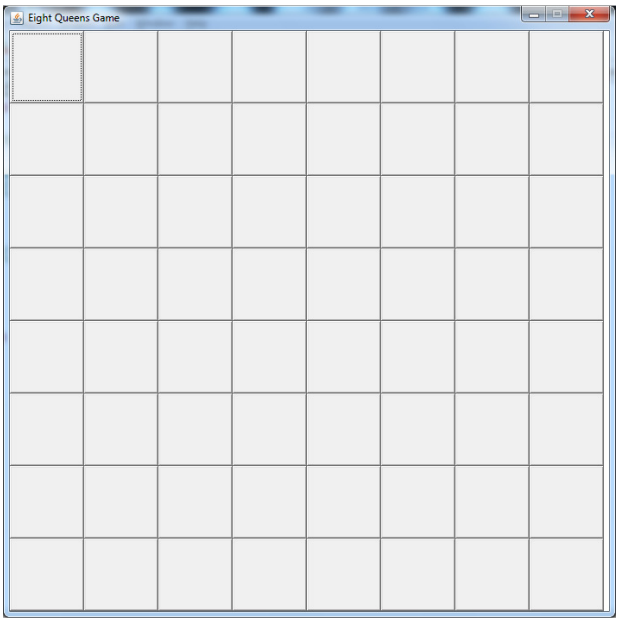
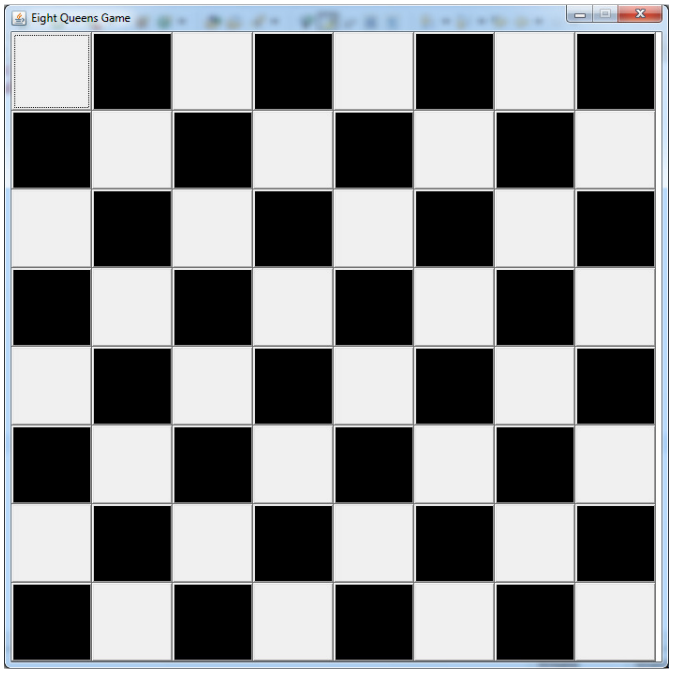
## The 8-Queens Game

**You have to place 8 Queens in such a way that no queen comes in the path of the other queen.**

1. Execute the program. You will see the following output.




2. Select the class ChessBoardRenderer. Write the logic for the method isBlackSquare(). Essentially this method takes the square on the board as a value. You need to determine if this square is a black square. If your logic is correct, then you will see the chess board appearing.



3. Select the class GameStrategy. Write the logic for verifying whether the square selected by the user is a valid position for placing the queen. Also keep track of the number of queens successfully placed by the user.
4. Write the logic for the method getCoulumn(). Essentially you need to figure out the column number (0 to 7) for the provided cellId (0-63).

```
private int getColumn(int cellId)
```

5. Write the logic for the method getRow().Essentially you need to figure out the row number (0 to 7) for the provided cellId (0-63).

```
private int getRow(int cellId)
```

6. Write the logic for the method isValidPosition().

```
public boolean isValidPosition(int cellId)
```

   a. You need to figure out if the selected cellId is a valid position to place the queen.
   b. You have to update the placedQueens array if the position (cellId) is valid.
   c. Increment the numQueens if the position is valid.

7. If you get your logic right, it is just about playing the game!