

Hausaufgabe 11.2: 8-Damen-Problem

```
public class EightQueens {

    static final int BOARDSize = 8;
    static int board[] = new int[BOARDSize];

    public static void main(String argv[]) {
        int i;

        // print ONE solution of the eight queens problem
        System.out.println("A solution to the eight queens problem is given by:\n");

        if (place(0)) {
            for (i=0;i<BOARDSize;++i) System.out.print((board[i]+1) + "\t");
            System.out.println("\nHere, the first number is the column for row 1,");
            System.out.println("The second number is the column for row 2, etc.");
        } else {
            System.out.println("Failed.");
        }
    } // end main

    // "safe" tests whether placement at (i,j) is compatible with rows 0 through i-1.
    static boolean safe(int i, int j) {
        int k = 0;
        boolean OK = true;

        while (OK && (k<i)) {
            if ((board[k] == j) || (board[k]-k == j-i) || (board[k]+k == i+j))
                OK = false;
            ++k;
        }
        return(OK);
    } // end of safe

    // Places pieces safely on rows i through BOARDSize, assuming
    // safe placement on rows 1 through i-1 has already been made.
    // Function returns "true" if this is possible, returns "false" else.
    static boolean place(int i) {
        int j = 0;
        boolean found = false;

        if (i == BOARDSize) {
            return(true);
        } else {
            while ((j < BOARDSize) && (!found)) {
                if (safe(i,j)) {
                    board[i] = j;
                    found = place(i+1);
                }
                ++j;
            }
            return(found);
        }
    } //end place
}
```