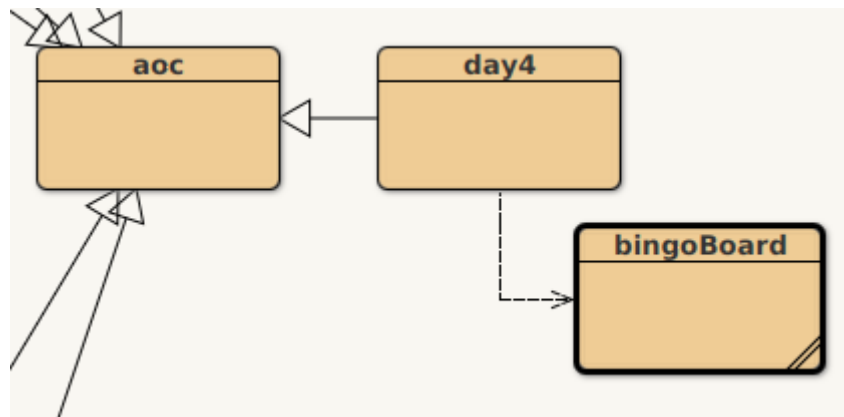


Tag 4: Bingo mit dem Riesenkraken

Hier kann man objektorientiert modellieren:



Man kann eine Klasse `bingoBoard` verwenden, um die Boards zu speichern, eine Methode `checkWin` kann dann zurückgeben, ob ein Board nach einer neuen Zahl gewonnen hat.

In `day4` hat mein zwei `ArrayLists` für die Bingozahlen und die Boards:

```
// day4: bingo numbers and bingoboards
ArrayList<Integer> bingoNumbers = new ArrayList<>();
ArrayList<bingoBoard> bingoBoards = new ArrayList<>();
```

Die `bingoBoard`-Klasse hat die Felder als zweidimensionales Array von `int`-Werten:

```
public class bingoBoard
{
    // Instanzvariablen - ersetzen Sie das folgende Beispiel mit Ihren Variablen
    private int[][] bField;
    private int boardSize = 5;
    private int winningNumber;

    /**
     * Konstruktor für Objekte der Klasse bingoBoard
     */
    public bingoBoard()
    {
        bField = new int[boardSize][boardSize];
    }
}
```

Außerdem sind Methoden wie

- `setBoardFieldValue(int x, int y, int value)`
- `checkWin()`
- `calcScore()`
- `markNumber(int number)`

möglicherweise hilfreich.

Das zentrale Problem ist die Verarbeitung des Inputs. Die erste Zeile muss in die bingoNumbers geparkt werden, und die weiteren Zeilen, getrennt durch Leerzeilen geben jeweils ein Board.

Codegerüst zur Verarbeitung des Inputs

```
public void parseInputToBoards () {
    this.bingoNumbers.clear();
    this.bingoBoards.clear();
    int lineNum = 0;

    bingoBoard b = null;
    int fRow = 0;
    int fCol = 0;

    for ( String[] line: input) {
        // Trim whitespaces
        line[0] = line[0].trim();
        // first line holds bingo numbers
        if (lineNum == 0) {
            String[] temp = line[0].split(",");
            for(String t: temp) {
                bingoNumbers.add(Integer.parseInt(t));
            }
            System.out.println("Bingo Numbers: " + bingoNumbers);
        } else {
            // Empty line starts new board
            if (line[0].trim().isEmpty()) {

                if ( b != null ) {
                    bingoBoards.add(b);
                }
                b=new bingoBoard();
                fRow = 0;
            } else {
                String[] temp = line[0].split("\\s+");
                fCol = 0;

                for (String t: temp) {
                    b.setBoardFieldValue(fCol, fRow, Integer.parseInt(t));
                    fCol++;
                }
                fRow++;
            }
        }
        lineNum++;
    }
}
```

```
// Add last board  
bingoBoards.add(b);  
}
```

From:
<https://www.info-bw.de/> -

Permanent link:
<https://www.info-bw.de/faecher:informatik:oberstufe:java:aoc:aoc2021:day4:start?rev=1638803557>

Last update: **06.12.2021 15:12**

