

# Konzept zu Aufgabe 02

## AufgabenStellung:

1. Verwirklichen von einem Sudoku-Editor mit einer Oberfläche als JTable, die ein Spielfeld bestehend aus 9 3x3 Matrizen besitzt.
2. In den Feldern sollen Werte von 1 bis 9 vorgegeben werden. Diese dürfen nicht verändert werden können.
3. Der eingetragene Wert darf nicht 2 mal in der zugehörigen Matrize als auch in der Spalte bzw. Zeile vorkommen.

## Ueberlegung:

### Das Model besitzt:

Felder:

- Ein Array in dem alle Werte zur Prüfung gespeichert werden
- Ein Array in dem festgehalten wird welche Zellen Editierbar sind
- Felder, die Anschnitte bzw. Groesse des Modells beschreiben

Methoden:

- Konstruktor, der das Modell mithilfe versch. Methoden initialisiert
- fillTable generiert zufällige Werte für das Modell.
- setEditable legt fest, ob eine Zelle editierbar ist oder nicht.
- getElements sammelt die aktuellen Werte der JTable
- solveGame speichert den Spielstand in einer Datei
- clearGame initialisiert das Modell und JTable neu
- isCellEditable prüft ob Zelle editierbar ist oder nicht

### Die View besitzt:

Felder:

- JComponents, welche die Darstellung eines Sudoku-Editor ermöglichen
- Eine Referenz auf das Modell (strong typed)

Methoden:

- Konstruktor, der die verschiedenen Methoden zur Erzeugung der View aufruft
- createEditArea erzeugt den editierbaren Abschnitt der View plus den Kopf
- createButtonArea erzeugt den Abschnitt mit Knöpfen die Aktionen auslösen
- createMainArea bringt alle Abschnitte zusammen und formt das Layout
- actionPerformed handelt ActionEvents von den Knöpfen
- main startet die Oberfläche

### Die Storage-Klasse besitzt:

- saveData diese Klassenmethode speichert ein 2 Dimensionales Array in eine Datei ab

### Anmerkungen:

Eine konsequente Implementierung der MVC – Architektur war mir durch einen fuer mich nicht vermeidbaren Stackoverflow nicht moeglich.  
(siehe //-- Kommentare im Code)

Falls Sie eine Loesung sehen ohne die ganze Struktur umzubauen waere ich dankbar fuer ein Feedback.