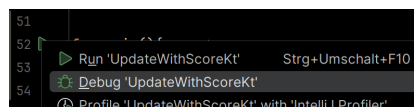


Führe die folgenden Schritte durch, um den Visual-Debugger in IntelliJ zu nutzen.

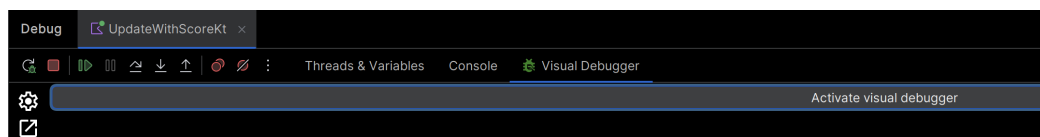
- Setze durch Klicks auf die Leiste neben dem Codefenster sogenannte *Breakpoints*. Beim Debuggen wird die Ausführung an diesen Stellen unterbrochen.

```
39 fun moveBoardLeft(model: GameState): GameState { 1 Max
40     var acc = emptyList<List<Int>>()
41     var score = model.score
42     for (x in model.board) {
43         val moveLeftResult = moveRowLeft(x)
44         acc = acc + listOf(moveLeftResult.row)
45         score += moveLeftResult.scoreIncrease
46     }
47     val newGameWon = model.gameWon || contains20(acc, number: 2048)
48     return GameState(acc, score, newGameWon)
49 }
50
51
52 fun main() { new *
53     println(moveBoardLeft(GameState(listOf(listOf(2, 0, 2), listOf(
54 }
```

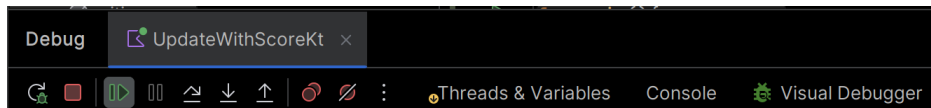
- Klicke mit der rechten Maustaste auf den Play-Button und anschließend auf Debug ...



- Klicke auf Activate Visual Debugger



- Wähle anschließend aus, dass du den Activate Visual Debugger im Browser öffnen willst.
- Platziere das Browserfenster und IntelliJ nebeneinander.
- Mit einem Klick auf den Resume-Button springst du zum nächsten Breakpoint.



- Im Browser siehst du immer das Objektdiagramm zum aktuellen Breakpoint. Die grünen Rechtecke stehen für Objekte, deren Attribute nicht ausschließlich zu primitiven Datentypen gehören. Die Attribute können durch einen Klick angezeigt werden.

