

Medium  3652  295  Add to List  Share

- Calculate the **sum** of the elements of `matrix` inside the rectangle defined by its **upper left corner** (`row1`, `col1`) and **lower right corner** (`row2`, `col2`).

- `NumMatrix(int[][] matrix)` Initializes the object with the integer matrix `matrix`.
- `int sumRegion(int row1, int col1, int row2, int col2)` Returns the **sum** of the elements of `matrix` inside the rectangle defined by its **upper left corner** (`row1, col1`) and **lower right corner** (`row2, col2`).

3	0	1	4	2
5	6	3	2	1
1	2	0	1	5
4	1	0	1	7
1	0	3	0	5

```
["NumMatrix", "sumRegion", "sumRegion", "sumRegion"]
```

```

1  ▼ class NumMatrix {
2
3      private int[][] dp;
4
5  ▼  public NumMatrix(int[][] matrix) {
6      if(matrix.length == 0 || matrix[0].length == 0)
7          return;
8
9      dp = new int[matrix.length + 1][matrix[0].length + 1];
10
11  ▼      for(int r = 0 ; r < matrix.length; r++){
12  ▼          for(int c = 0 ; c < matrix[0].length; c++){
13              dp[r + 1][c + 1] = dp[r + 1][c] + dp[r][c + 1] + matrix[r]
[c] - dp[r][c];
14          }
15      }
16  }
17
18  ▼  public int sumRegion(int row1, int col1, int row2, int col2) {
19      return dp[row2 + 1][col2 + 1] - dp[row1][col2 + 1] - dp[row2 + 1]
[col1] + dp[row1][col1];
20  }
21  }
22
23  ▼ /**
24   * Your NumMatrix object will be instantiated and called as such:
25   * NumMatrix obj = new NumMatrix(matrix);
26   * int param_1 = obj.sumRegion(row1,col1,row2,col2);
27   */

```

Your previous code was restored from your local storage. [Reset to default](#)