AlgoExpert | Quad Layout | Java | 14px | Sublime | Monokai | 00:00:0

Our Solution(s) Run Code

Solution 1 Solution 2

```java
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

class Program {
  // O(nk) time | O(n) space
  public static int maxProfitWithKTransactions(int[] prices, int k) {
    if (prices.length == 0) {
      return 0;
    }
    int[] evenProfits = new int[prices.length];
    int[] oddProfits = new int[prices.length];
    for (int i = 0; i < prices.length; i++) {
      evenProfits[i] = 0;
      oddProfits[i] = 0;
    }
    for (int t = 1; t < k + 1; t++) {
      int maxThusFar = Integer.MIN_VALUE;
      int[] currentProfits = new int[prices.length];
      int[] previousProfits = new int[prices.length];
      if (t % 2 == 1) {
        currentProfits = oddProfits;
        previousProfits = evenProfits;
      } else {
        currentProfits = evenProfits;
        previousProfits = oddProfits;
      }
      for (int d = 1; d < prices.length; d++) {
        maxThusFar = Math.max(maxThusFar, previousProfits[d - 1] - pri
        currentProfits[d] = Math.max(currentProfits[d - 1], maxThusFar
      }
    }
    return k % 2 == 0 ? evenProfits[prices.length - 1] : oddProfits[pr
  }
}
```

Your Solutions Run Code

Solution 1 Solution 2 Solution 3

```java
class Program {
  public static int maxProfitWithKTransactions(int[] prices, int k) {
    // Write your code here.
    return -1;
  }
}

```

Our Tests

[illegible]

Custom Output Submit Code

```
[illegible]
```

Run or submit code when you're ready.