

Benewnde Pierre BONKOUNGOU

Programming Assignment Unit 4

Code

```
/**
 * StockAnalysis class provides methods for analyzing stock prices.
 */

import java.util.ArrayList;

public class StockAnalysis {

    /**
     * Calculates the average stock price for both an array and an
     ArrayList.
     *
     * @param prices Array of stock prices.
     * @return Average stock price.
     */
    public static float calculateAveragePrice(float[] prices) {
        float sum = 0;
        for (float price : prices) {
            sum += price;
        }
        return sum / prices.length;
    }

    /**
     * Calculates the average stock price for an ArrayList.
     *
     * @param prices ArrayList of stock prices.
     * @return Average stock price.
     */
    public static float calculateAveragePrice(ArrayList<Float> prices)
{
        float sum = 0;
        for (float price : prices) {
            sum += price;
        }
        return sum / prices.size();
    }

    /**
     * Finds the maximum stock price for an array.
     *
     * @param prices Array of stock prices.
     */
}
```

```

    * @return Maximum stock price.
    */
    public static float findMaximumPrice(float[] prices) {
        float maxPrice = prices[0];
        for (float price : prices) {
            if (price > maxPrice) {
                maxPrice = price;
            }
        }
        return maxPrice;
    }

    /**
     * Finds the maximum stock price for an ArrayList.
     *
     * @param prices ArrayList of stock prices.
     * @return Maximum stock price.
     */
    public static float findMaximumPrice(ArrayList<Float> prices) {
        float maxPrice = prices.get(0);
        for (float price : prices) {
            if (price > maxPrice) {
                maxPrice = price;
            }
        }
        return maxPrice;
    }

    /**
     * Counts the occurrences of a specific price in an array.
     *
     * @param prices      Array of stock prices.
     * @param targetPrice Target stock price to count occurrences.
     * @return Number of occurrences.
     */
    public static int countOccurrences(float[] prices, float
targetPrice) {
        int count = 0;
        for (float price : prices) {
            if (price == targetPrice) {
                count++;
            }
        }
        return count;
    }

    /**
     * Counts the occurrences of a specific price in an ArrayList.
     *
     * @param prices      ArrayList of stock prices.

```

```

    * @param targetPrice Target stock price to count occurrences.
    * @return Number of occurrences.
    */
    public static int countOccurrences(ArrayList<Float> prices, float
targetPrice) {
        int count = 0;
        for (float price : prices) {
            if (price == targetPrice) {
                count++;
            }
        }
        return count;
    }

    /**
     * Computes the cumulative sum of stock prices for an ArrayList.
     *
     * @param prices ArrayList of stock prices.
     * @return ArrayList containing cumulative sum of prices.
     */
    public static ArrayList<Float>
computeCumulativeSum(ArrayList<Float> prices) {
        ArrayList<Float> cumulativeSum = new ArrayList<>();
        float sum = 0;
        for (float price : prices) {
            sum += price;
            cumulativeSum.add(sum);
        }
        return cumulativeSum;
    }

    /**
     * Main method for testing the StockAnalysis methods.
     *
     * @param args Command line arguments (not used).
     */
    public static void main(String[] args) {
        // Example usage:
        float[] stockPricesArray = {10.5f, 12.3f, 15.2f, 14.8f, 11.7f,
13.5f, 14.2f, 12.6f, 10.9f, 11.8f};
        ArrayList<Float> stockPricesArrayList = new ArrayList<>();
        for (float price : stockPricesArray) {
            stockPricesArrayList.add(price);
        }

        // Print results for the array
        System.out.println("Average Stock Price (Array): " +
calculateAveragePrice(stockPricesArray));
        System.out.println("Maximum Stock Price (Array): " +
findMaximumPrice(stockPricesArray));
    }

```

```

        System.out.println("Occurrences of 14.2 (Array): " +
countOccurrences(stockPricesArray, 14.2f));

        // Print results for the ArrayList
        System.out.println("\nAverage Stock Price (ArrayList): " +
calculateAveragePrice(stockPricesArrayList));
        System.out.println("Maximum Stock Price (ArrayList): " +
findMaximumPrice(stockPricesArrayList));
        System.out.println("Occurrences of 14.2 (ArrayList): " +
countOccurrences(stockPricesArrayList, 14.2f));

        // Compute and print cumulative sum for the ArrayList
        System.out.println("\nCumulative Sum of Stock Prices
(ArrayList): " + computeCumulativeSum(stockPricesArrayList));
    }
}

```

Explanation:

1. **calculateAveragePrice Method:**
 - Calculates the average stock price for both an array and an ArrayList.
 - Uses a loop to iterate through each element in the array or ArrayList, adds the prices, and then calculates the average.
1. **findMaximumPrice Method:**
 - Finds the maximum stock price for both an array and an ArrayList.
 - Initializes maxPrice with the first element, then iterates through the array or ArrayList to find the maximum.
2. **countOccurrences Method:**
 - Counts the occurrences of a specific price in both an array and an ArrayList.
 - Uses a loop to iterate through each element and increments count when the target price is found.
3. **computeCumulativeSum Method:**
 - Computes the cumulative sum of stock prices for an ArrayList.
 - Uses a loop to iterate through each price, adds it to the running sum, and stores the cumulative sum in an ArrayList.
4. **main Method:**
 - Provides an example of how to use the methods with both arrays and ArrayLists.
 - Creates an array and an ArrayList of stock prices for demonstration purposes.
 - Prints the results for average, maximum, and occurrences for both the array and ArrayList.
 - Computes and prints the cumulative sum for the ArrayList.

Output:

```
C:\Users\benew\OneDrive\UoPeople\MATH 1280 - Introduction to Statistics\UNIT_4-DR_Variables>java StockAnalysis
Average Stock Price (Array): 12.75
Maximum Stock Price (Array): 15.2
Occurrences of 14.2 (Array): 1

Average Stock Price (ArrayList): 12.75
Maximum Stock Price (ArrayList): 15.2
Occurrences of 14.2 (ArrayList): 1

Cumulative Sum of Stock Prices (ArrayList): [10.5, 22.8, 38.0, 52.8, 64.5, 78.0, 92.2, 104.799995, 115.7, 127.5]
```

References

- Oracle Documentation for Java (n.d.). Retrieved from <https://docs.oracle.com/en/java/>
- Java ArrayList Documentation (n.d.). Retrieved from <https://docs.oracle.com/en/java/javase/15/docs/api/java.base/java/util/ArrayList.html>
- GeeksforGeeks - Java.util.Arrays Class in Java (n.d.). Retrieved from <https://www.geeksforgeeks.org/java-util-arrays-class-java/>