This program is not an example of good program design. Rather, its purpose is to show various ways of creating and using arrays. The arrays `monthName` and `rainfall` are parallel arrays because each has length 12 and their corresponding entries are related. That is, `monthName[0]` is January and `rainfall[0]` is the January rainfall.

```java
import java.text.*;
import javax.swing.*;

/* @author Dr. Caffeine
 * There are a few changes in this Chapter 10 program from your text:
 *    Input is done using a Scanner object
 *    Formatted output of decimal numbers is done using
 *       printf instead of using a DecimalFormat object
 */
import java.util.Scanner;

class Ch10RainfallStat
{
    public static void main (String[] args)
    {
        Scanner input = new Scanner(System.in);

        String[] monthName = {
            "August", "September", "October", "November", "December" 
        };

        double[] rainfall = new double[12];
        double[] quarterAverage = new double[4];

        double annualAverage, sum, oddMonthSum, oddMonthAverage,
                evenMonthSum, evenMonthAverage;

        ///// Read in the monthly rainfall calculate annual average /////
        sum = 0.0;
        for (int i = 0; i < rainfall.length; i++)
        {
            System.out.print("Rainfall for " + monthName[i] + ": ");
            rainfall[i] = input.nextDouble();
            sum += rainfall[i];
        }

        annualAverage = sum / 12.0;
        System.out.printf("\n\nAnnual Average Rainfall: %06.2f\n\n", annualAverage);

        ///// Odd and Even Month Averages /////
        oddMonthSum = evenMonthSum = 0.0;
        for (int i = 0; i < rainfall.length; i += 2)
        {
            oddMonthSum += rainfall[i];
        }
        oddMonthAverage = oddMonthSum / 6.0;
    }
}
```
for (int i = 1; i < rainfall.length; i += 2) {
    evenMonthSum += rainfall[i];
}
evenMonthAverage = evenMonthSum / 6.0;

/***** As an alternative, find even and odd sums in a single loop*****/
/***** Three variables are declared and initialized.  *****/for (int i = 0, oddSum = 0, evenSum = 0; i < rainfall.length; i += 2 ) {
    oddSum += rainfall[i];
    evenSum += rainfall[i+1];
}
System.out.println("Odd & Even Month Rainfall Averages:");
System.out.printf(    " Odd Month Average: %-12.2f inches\n", oddMonthAverage);
System.out.printf(    " Even Month Average: %012.2f inches\n", evenMonthAverage);

/// Quarter Averages ///
System.out.println("Quarterly Rainfall Averages:");
for (int i = 0; i < 4; i++){
    sum = 0;
    for (int j = 0; j < 3; j++) //compute the sum of
    {
        sum += rainfall[3*i + j]; //one quarter
    }
    quarterAverage[i] = sum / 3.0; //average for Quarter i+1
    System.out.printf(    " Average Qtr %d: %6.5f inches\n", (i+1), quarterAverage[i]);
}

---

**The Input:**

Rainfall for January: 0
Rainfall for February: 0.5
Rainfall for March: 1.2
Rainfall for April: 8.3
Rainfall for May: 18.2
Rainfall for June: 5.6
Rainfall for July: 2.1
Rainfall for August: 1.3
Rainfall for September: 1
Rainfall for October: .3
Rainfall for November: 0
Rainfall for December: 0

**The Output:**

Annual Average Rainfall: 003.21

Odd & Even Month Rainfall Averages:
    Odd Month Average: 3.75 inches
    Even Month Average: 00000002.67 inches

Quarterly Rainfall Averages:
    Average Qtr 1: 0.56667 inches
    Average Qtr 2: 10.70000 inches
    Average Qtr 3: 1.46667 inches
    Average Qtr 4: 0.10000 inches
What is an alternative to using parallel arrays? Use an object to store the related data:

<table>
<thead>
<tr>
<th>MonthlyRain</th>
</tr>
</thead>
<tbody>
<tr>
<td>· month (String)</td>
</tr>
<tr>
<td>· rainfall (double)</td>
</tr>
</tbody>
</table>

+ MonthlyRainfall(String, double)
+ MonthlyRainfall(String)

+ String getMonth()
+ double getRainfall()
+ void setRainfall(double)
+ String toString()

Then, what type of array would we use?

```java
MonthlyRain[] data = new MonthlyRain[12];
data[0] = new MonthlyRain("January");
...
```

OR

```java
```

OR

```java
String[] month = {"January", "February", ..., "December"}
for(int m = 0; m < 12; m++)
{
    data[m] = new MonthlyRain(month[m]);
}
```