**Design:**

**Data types used:**

Float array:

monthlyTotalRain , monthlyHighestTemp

A floats array is used to store the different readings, floats are used since it creates a greater degree of accuracy, allowing the user input accurate readings. Using arrays allows me to store multiple data in one variable, this will be useful when storing the multiple data, thus will save time and data. Using arrays means I do not need to store data into multiple variables. Having an array also makes it easier to make loops.

Integer:

Month

Integers are whole numbers, this means using this data type will be useful especially for recording months. This is also more efficient compared to float, since it requires less memory and also runs faster when called from.

Pseudo Code:

Import scanner

Set global montlyTotalRain as float array

Set global monthle highestTemp as float array

Public Static void main

Set sc as scanner

Set keyboard as scanner

For loop(I = 1,I <13,+1)

monthlyTotalRain[i] = validate(0,300)

montlyHighestTemp[i] = validate(-30,60)

int choice = 1

while choice != 0

displayMenu()

choice = input(“enter choice”)

if choice = 1

displayData()

if choice = 2

month = input(“enter month”)

nTemp = input(“new temp:”)

changeTemp(month,nTemp)

if choice = 3

month = input(“enter month”)

nRain= input(“new temp:”)

changeTemp(month,nRain)

if choice = 5

print(getMean(monthleHighestTemp),getMean(monthlyTotalRain))

if choice = 6

displayLowerThan(monthlyHighestTemp,3)

if choice = 7

displayLowerThan(returnMeanArray(monthlyTotalRain),0.75)

if choice = 8

displayHighestLowest(monthlyHighestTemp)

if choice = 9

display highestLowest(monthlyTotalRain)

public static void menu

print(menu)

public static float validate(min,max)

temp = input(“new temp:”)

valid = 0

while (valid != 1)

if min <temp < max

valid = 1

return temp

public static void displayData()

for i in range 0 to 12

print(monthlyhighestTemp(i),monthlyTotalRain(i))

changeTemp(month, change)

monthlyhighestTemp(month) = change

changerain(month,change)

monthlyTotalRain(month ) = change

getTotal(float[] weatherArray)

total = 0

for I in range 0 to 12

total += weatherArray[i]

return total

getMean(float[] weatherArray)

float mean = gettotal(weatherArray)/ weatherArray.length

displayLowerThan(float[] weatherArray,lowest)

for I in tange 0 to 12

if weatherArray(i) < lowest

pribnt(weatherArray(i))

displayHighestLowest(float[] weatherArray)

lowest = 0

highet = 0

for I in range 0 to 12

if weaherArray[i] < weatherArray[lowest]

lowest = i

if weatherArray[i] > weatherArray[highest]

highest = i

print(weatherArray(lowest),weatherArray(highest))

**Testing:**

The way I am testing the program, is to input data, predict the output, and if the actual output is the same as dicted output then the code is working.

**Entered data**

**Table 1 -**

|  |  |  |
| --- | --- | --- |
| **Month** | **Total rainfall** | **Highest temperature** |
| **1** | 53.6 | 8.4 |
| **2** | 46.2 | 7.5 |
| **3** | 25.6 | 11.4 |
| **4** | 22.5 | 15.9 |
| **5** | 55.6 | 16.8 |
| **6** | 25.2 | 21.2 |
| **7** | 47.8 | 22.6 |
| **8** | 57.2 | 21.5 |
| **9** | 52 | 18 |
| **10** | 35 | 15.1 |
| **11** | 65.3 | 13.1 |
| **12** | 62.9 | 13.6 |

**Table 2**

|  |  |  |
| --- | --- | --- |
| **Month** | **Total rainfall** | **Highest temperature** |
| **1** | 53.6 | 8.4 |
| **2** | 46.2 | 7.5 |
| **3** | 25.6 | 11.4 |
| **4** | 22.5 | 15.9 |
| **5** | 55.6 | 16.8 |
| **6** | 25.2 | 21.2 |
| **7** | 47.8 | 22.6 |
| **8\*** | .2 | -2 |
| **9** | 52 | 18 |
| **10** | 35 | 15.1 |
| **11** | 65.3 | 13.1 |
| **12** | 62.9 | 13.6 |

|  |  |  |  |
| --- | --- | --- | --- |
| Description | Data entered | Expected result | Actual result |
| Test invalid rainfall input | -1 | Error message | -1  Invalid reading, enter valid reading: |
|  | 301 | Error message | 301  Invalid reading, enter valid reading: |
|  | 2334 | Error message | 2334  Invalid reading, enter valid reading: |
| Test invalid temperature | -31 | Error message | Enter the highest temperature for month 1:  -31 |
|  | 61 | Error message | Invalid reading, enter valid reading:  61 |
| Enters and saves data then display readings | Table 1 | Data is identical to table 1 | Month 01; Total rainfall: 53.6; Highest temperature: 8.4  Month 11; Total rainfall: 46.2; Highest temperature: 7.5  Month 21; Total rainfall: 25.6; Highest temperature: 11.4  Month 31; Total rainfall: 22.5; Highest temperature: 15.9  Month 41; Total rainfall: 55.6; Highest temperature: 16.8  Month 51; Total rainfall: 25.2; Highest temperature: 21.2  Month 61; Total rainfall: 47.8; Highest temperature: 22.6  Month 71; Total rainfall: 57.2; Highest temperature: 21.5  Month 81; Total rainfall: 52.0; Highest temperature: 18.0  Month 91; Total rainfall: 35.0; Highest temperature: 18.0  Month 101; Total rainfall: 65.3; Highest temperature: 13.1  Month 111; Total rainfall: 62.9; Highest temperature: 13.6   * Error found, month number display wrong number, this has been solved by saving the month number as a separate variable instead of “I + 1” in print statement |
| Display annual rainfall total | Table 1 | 548.9 | Enter your choice:5  The total annual rainfall is 548.9 |
| Display mean temperature and mean rainfall | Table 1 | Rainfall : 45.74  Temperature :15.67 | The mean rainfall is 45.74167, The mean highest temperature is 15.666667 |
| Display temp below 3 degrees | Table 1 | error message, no month fit crieria | Enter your choice:7  Months with highest temperature below 3 degrees:  No month with reading lower than3.0 |
|  | Table 2 | Display month 8 | Months with highest temperature below 3 degrees:  Month 8 with -2.0 |
| Display month with avg daily rainfall in drought | Table 1 | Error message, no month fit | Enter your choice:8  Months below drought level(0.075mm):  No month with reading lower than0.75 |
|  | Table 2 | Display month 8 | Enter your choice:8  Months below drought level(0.075mm):  Month 8 with 0.006666667 |
| Display highest temp and lowest temp | Table 1 | Lowest: month 2  Highest: month 7 | Month 2 has the lowest reading of 7.5,Month 7 has the highest reading of 22.6 |
| Display driest and wettest month | Table 1 | Driest: month 4  Wettest : month 11 | Month 4 has the lowest reading of 22.5,Month 11 has the highest reading of 65.3 |
| Change temperature and rainfall | Original: table 1  Change :month 8  Temp: -3  Rainfall: | Displays data in table | Original:  Month 1; Total rainfall: 53.6; Highest temperature: 8.4  Month 2; Total rainfall: 46.2; Highest temperature: 7.5  Month 3; Total rainfall: 25.6; Highest temperature: 11.4  Month 4; Total rainfall: 22.5; Highest temperature: 15.9  Month 5; Total rainfall: 55.6; Highest temperature: 16.8  Month 6; Total rainfall: 25.2; Highest temperature: 21.2  Month 7; Total rainfall: 47.8; Highest temperature: 22.6  Month 8; Total rainfall: 57.2; Highest temperature: 21.5  Month 9; Total rainfall: 52.0; Highest temperature: 18.0  Month 10; Total rainfall: 35.0; Highest temperature: 18.0  Month 11; Total rainfall: 65.3; Highest temperature: 13.1  Month 12; Total rainfall: 62.9; Highest temperature: 13.6  New  Enter your choice:2  Month 1; Total rainfall: 53.6; Highest temperature: 8.4  Month 2; Total rainfall: 46.2; Highest temperature: 7.5  Month 3; Total rainfall: 25.6; Highest temperature: 11.4  Month 4; Total rainfall: 22.5; Highest temperature: 15.9  Month 5; Total rainfall: 55.6; Highest temperature: 16.8  Month 6; Total rainfall: 25.2; Highest temperature: 21.2  Month 7; Total rainfall: 47.8; Highest temperature: 22.6  Month 8; Total rainfall: 0.2; Highest temperature: -2.0  Month 9; Total rainfall: 25.0; Highest temperature: 18.0  Month 10; Total rainfall: 35.0; Highest temperature: 15.1  Month 11; Total rainfall: 65.3; Highest temperature: 13.1  Month 12; Total rainfall: 62.9; Highest temperature: 13.6 |

Source code:

package weatheranalysis;

/\*Imports scanner tool\*/

import java.util.Scanner;

/\*\*

\*

\* @author christian

\*/

public class WeatherAnalysis {

/\*creates array to store weather readings\*/

public static float[] monthlyTotalRain = new float[12];

public static float[] monthlyHighestTemp = new float[12];

public static void main(String[] args){

Scanner sc = new Scanner(System.in);

Scanner keyboard = new Scanner(System.in);

/\*Loop used to store reading into array, for month order\*/

/\*counter used to determine location to add reading\*/

int counter = 0;

for (int month = 1; month < 13;month ++){

counter = month -1;

System.out.println("Enter the total rainfall for month "+ month + ":");

/\*totaRain uses validation method to make sure data entered is in the right range\*/

/\*Place data entered to right location in array\*/

float totalRain = validate(0,300);

monthlyTotalRain[counter] = totalRain;

System.out.println("Enter the highest temperature for month "+ month + ":");

/\*totaRain uses validation method to make sure data entered is in the right range\*/

/\*Place data entered to right location in array\*/

float highestTemp = validate(-30,60);

monthlyHighestTemp[counter] = highestTemp;

}

System.out.println("Data entered");

/\*Loop acts as main menu to code, each input leads to method\*/

int exit = 1;

int choice;

while (exit != 0){

/\*runs the displayMenu method\*/

displayMenu();

System.out.print("Enter your choice:");

choice = keyboard.nextInt();

switch (choice){

/\*Sets exit to 0, this exits program\*/

case 1:

exit = 0;

break;

case 2:

/\*Displays readings by running displayData method\*/

displayData();

break;

case 3:

/\*Asks user for new month,used to determine location in array to place new reading

\*Ask user for new reading

\*uses validate method to make sure data enter is valid

\*runs change reading function

\*/

System.out.println("Enter the month you want to change:");

int month = (int)validate(1,12);

System.out.println("Enter new temperature:");

float nTemp = validate(-30,60);

changeTemp(month,nTemp);

break;

case 4:

/\*Asks user for new month,used to determine location in array to place new reading

\*Ask user for new reading

\*uses validate method to make sure data enter is valid

\*runs change reading function

\*/

System.out.println("Enter the month you want to change:");

int nMonth = (int)validate(1,12);

System.out.println("Enter new total rainfall:");

float nRain = validate(0,300);

changeRain(nMonth,nRain);

break;

case 5:

/\*runs getTotal to find total annual rainfall\*/

System.out.println("The total annual rainfall is "+ getTotal(monthlyTotalRain));

break;

case 6:

/\*uses getMean to calculate mean rainfall\*/

System.out.println("The mean rainfall is "+ getMean(monthlyTotalRain)+", The mean highest temperature is "+ getMean(monthlyHighestTemp));

break;

case 7:

/\*displays temperature elow 3 degrees\*/

System.out.println("Months with highest temperature below 3 degrees:");

displayLowerThan(monthlyHighestTemp,3);

break;

case 8:

/\*Displays months that are below daily drought\*/

System.out.println("Months below drought level(0.075mm):");

displayLowerThan(returnMeanArray(monthlyTotalRain), (float) 0.75);

break;

case 9:

/\*Displays highest and lowest readings\*/

displayHighestLowest(monthlyHighestTemp);

break;

case 10:

/\*Displays highest and lowest readings\*/

displayHighestLowest(monthlyTotalRain);

break;

default:

/\*Displays error message if option entered is invalid\*/

System.out.println("Incorrect input");

}

}

}

public static void displayMenu(){

/\*Displays menu option\*/

System.out.println("1. Quit \n 2. Display all Data \n 3. Change a temperature value \n 4. Change a rainfall value \n 5. Display annual rainfall total \n 6.Display mean temperature and mean total rainfall \n 7. Display month with temp below 3 degrees \n 8. Display month below drought level \n 9. Display month(s) with highest temp and lowest temp \n 10. Display driest and wettest month");

}

public static float validate(float min, float max){

/\*valid variable used to exit loop if data is valid

\*scanner retreives and saves data enered

\*loop until data entered is valid

\*makes sure data entred does not exceed ranges

\*/

int valid = 0;

Scanner sc = new Scanner(System.in);

float temp = sc.nextFloat();

while (valid != 1){

if(temp < min || temp > max){

System.out.println("Invalid reading, enter valid reading:");

temp = sc.nextFloat();

}

else{

valid = 1;

}

}

return temp;

}

public static void displayData(){

/\*Loop 12 times, to display readings for each month\*/

for (int i = 0; i < 12; i++){

int month = i + 1;

System.out.println("Month "+ month +"; Total rainfall: "+ monthlyTotalRain[i]+"; Highest temperature: "+ monthlyHighestTemp[i] );

}

}

public static void changeTemp(int month, float change){

/\*Change reading in array to new reading\*/

monthlyHighestTemp[month - 1] = change;

}

public static void changeRain(int month, float change){

/\*Change reading in array to new reading\*/

monthlyTotalRain[month - 1] = change;

}

public static float getTotal(float[] weatherArray){

/\*Loop to add each data in array to calculate total\*/

float total = 0;

for (int month = 0;month < 12;month++){

total += weatherArray[month];

}

return total;

}

public static float getMean(float[] weatherArray){

/\*Uses get total, then divides it with the weather array length to return mean\*/

float mean = getTotal(weatherArray)/weatherArray.length;

return mean;

}

public static float[] returnMeanArray (float[] weatherArray){

/\*Returns array to with mean for each month\*/

float[] meanArray = new float[12];

for (int i = 0; i < 12; i++){

meanArray[i] = weatherArray[i]/30;

}

return meanArray;

}

public static void displayLowerThan(float[] weatherArray,float lowest){

/\*Checks every data in array, then returns if reading is lower than lowest range entered\*/

int counter = 0;

for (int month = 0;month < 12;month ++){

if (weatherArray[month] < lowest){

int displayMonth = month + 1;

System.out.println("Month " + displayMonth +" with " + weatherArray[month]);

counter +=1;

}

}

if (counter == 0){

System.out.println("No month with reading lower than" + lowest);

}

}

public static void displayHighestLowest(float[] weatherArray){

/\*Checks all readings in array, if it is lower or higher than current highest or lowest, it is saved then displayed\*/

int lowestMonth = 0;

int highestMonth = 0;

for (int month = 1; month < 12; month ++){

if (weatherArray[month]< weatherArray[lowestMonth]){

lowestMonth = month;

}

if (weatherArray[month]> weatherArray[highestMonth]){

highestMonth = month;

}

}

int displayLowestMonth =lowestMonth+ 1;

int displayHighestMonth = highestMonth + 1;

System.out.println("Month "+ displayLowestMonth +" has the lowest reading of "+ weatherArray[lowestMonth]+",Month "+ displayHighestMonth +" has the highest reading of "+ weatherArray[highestMonth]);

}

}