

الجمهورية العربية السورية
المعهد العالي للعلوم التطبيقية والتكنولوجيا
قسم المعلومات
السنة الرابعة

Miniproject Java

اشراف

د. زهير دحروج

م. غيث منصور

م. نور نصار

تقديم الطالبين:

رغد ابراهيم

رامي شحادة

شرح العمل المنجز:

بدايةً، قمنا بإنشاء صف `ReadFiles` يسمح لنا باستخراج المعلومات من الملف المدخل باستخدام عدة توابع سنأتي على ذكرها.

- التابع `GetAddress(String path)`: يسمح لنا هذا التابع باستخراج عنوان ملف الدخل والموجود في أول سطر من الملف.

```
public String GetAddress (String path) throws FileNotFoundException, IOException{
    //Get the address of the text file (the first line)

    BufferedReader br = new BufferedReader(new FileReader(path));
    String address=br.readLine();
    br.close();
    return address;
}
```

- التابع `GetNumberOfYears(String path)`: يسمح لنا هذا التابع بمعرفة عدد السنوات التي تم تسجيل القيم فيها لكل مدينة. وهو فرضاً عدد ثابت من أجل كل سنة.

```
public int GetNumberOfYears (String path) throws FileNotFoundException, IOException
{
    //gets the number of the years which is constant in all the citys

    BufferedReader br = new BufferedReader(new FileReader(path));
    int n=0;
    br.readLine();//the address
    br.readLine();//the first city
    while(br.readLine().charAt(4)=='(',')')//for example "2004, 256": ',' is the forth character
    {
        n++;
    }
    br.close();
    return n;
}
```

- التابع GetNumberOfCities(String path): يسمح لنا هذا التابع بمعرفة عدد المدن الكلي في الملف.

```
public int GetNumberOfCities(String path) throws IOException,FileNotFoundException
{
    //gets the number of cities (maximum 10)
    BufferedReader br = new BufferedReader(new FileReader(path));
    int n=0;
    br.readLine();//the address
    String cl = br.readLine();//the first city
    while(cl!=null)
    {
        if (cl.charAt(0) >=65) //the first character is in the alphabet (A=65 in ASCII)
        {
            cl = br.readLine();//we read the city
            n++;//increase number of cities by one
        }
        else
            cl = br.readLine(); //we move on because this line is a year with the rain rate of this year
    }
    br.close();
    return n;
}
```

- التابع GetCities(String path): يعيد هذا التابع مصفوفة من نمط String تحوي على أسماء المدن الموجودة في الملف.

```
public String [] GetCities(String path) throws FileNotFoundException, IOException
{
    //gets the names of the cities
    int cityNum =0;
    cityNum=GetNumberOfCities(path);//number of cities
    String[] c=new String[cityNum];//allocate an array of Strings
    BufferedReader br = new BufferedReader(new FileReader(path));//open the file
    br.readLine();//read the address
    int j=0;
    String cl = br.readLine();//the first city
    while(cl!=null)
    {
        if (cl.charAt(0)>=65)
        {
            c[j]=cl;//save the city in the index j
            cl=br.readLine();//move on
            j++;
        }
        else
            cl = br.readLine(); //we move on because this line is a year with the rain rate of this year
    }
    br.close();
    return c;
}
```

- التابع (GetAves(String path): يعيد هذا التابع مصفوفة من نمط int, تحوي على متوسط معدلات الامطار في كل مدينة.

```
public int [] GetAves(String path) throws FileNotFoundException, IOException, NullPointerException
{
    //gets the avarege of each city
    BufferedReader br = new BufferedReader(new FileReader(path)); //open the file
    int nc=GetNumberOfCities(path); //number of citys
    int ny=GetNumberOfYears(path); //number of years
    int[] citiesAvg = new int[nc]; //allocate an array of the all city's avarege
    String cl=br.readLine(); //the address
    int sum=0; //the sum of the rain rate in the years in a city
    for(int k=0;k<nc;k++)
    {
        sum=0;
        cl=br.readLine(); //the first city
        {
            for(int i=0;i<ny;i++)
            {
                cl=br.readLine();
                sum+=Integer.parseInt(cl.substring(6)); //we take the avarege the ith year and parse it to integar
                //for example "2004, 456" the element in 6 is 4 and we take the substring from 6 to the end of this String
                //and parse it to integar for sum it with the other years
            }
            citiesAvg[k]=(sum/ny); //we calculate the avarege of all the years of this city
        }
        //move to the next city if we didn't reach all cities
    }
    br.close();
    return citiesAvg;
}
```

- التابع `getYearsOfcity (String city , String path)` يعيد هذا التابع مصفوفة من نمط `int`. تخزن السنوات التي تم تسجيل القيم فيها في المدينة المطلوبة.

```
public int [] getYearsOfcity (String city , String path) throws IOException
{
    BufferedReader br = new BufferedReader(new FileReader(path));
    br.readLine();//title
    int ny=GetNumberOfYears(path);
    int [] years =new int[ny];
    String c=br.readLine();
    while( c!=null){

        if(c.equals(city)) //we reach the city we want
        {
            int index=0;
            while(index!=ny)//put all the years in an array and returns it
            {
                String y=br.readLine();
                y = y.substring(0, 4);
                int year=Integer.parseInt(y);
                years[index]=year;
                index++;
            }
            break;
        }
        else c=br.readLine();
    }

    return years;
}
```

- التابع (String city , String path) getRatesOfcity: يعيد مصفوفة من نمط int. تحتوي على معدلات هطول الأمطار في كل سنة من السنوات في المدينة الواحدة.

```
public int [] getRatesOfcity (String city , String path) throws IOException
{
    BufferedReader br = new BufferedReader(new FileReader(path));
    String cl;
    cl=br.readLine();//title
    int ny=GetNumberOfYears(path);
    int [] Rates =new int[ny];
    cl=br.readLine();
    while(cl!=null){

        if(cl.equals(city)) //we reach the city we want
        {
            int index=0;
            while(index!=ny)//put all the rain Rates in an array and returns it
            {
                String y=br.readLine();
                y = y.substring(6);
                int year=Integer.parseInt(y);
                Rates[index]=year;
                index++;
            }
            break;
        }
        else cl=br.readLine();
    }
    return Rates;
}
```

- التابع (String path) getFactorOfRates: يعيد هذا التابع قيمة من النمط int والتي سوف تساعدنا في اختيار مقياس رسم مناسب لل histogram. من أجل قيمة ابتدائية لل factor=1 يقوم هذا التابع بالمرور على كل معدلات السنين، ومن أجل كل معدل نقارن حاصل قسمة المعدل نفسه على factor .

إذا كان حاصل القسمة أكبر من 900 والذي يمثل فرضاً طول الشاشة سنقوم بزيادة قيمة ال factor بمقدار 1.

```
public int getFactorOfRates(String path) throws IOException
{
    int[] Rates;
    int factor=1;
    String [] cities =GetCities(path);
    for(String city:cities)
    {
        Rates=getRatesOfcity(city,path);
        for(int l:Rates)
        {
            while((l/factor)>900)
                factor++;
        }
    }
    return factor;
}
```

كما قمنا بإنشاء الصف Draw الذي يحوي التابع التالي:

التابع (drawLines (Graphics g , int factor) : الذي قمنا باستخدامه لرسم
تدرجات ال histogram.

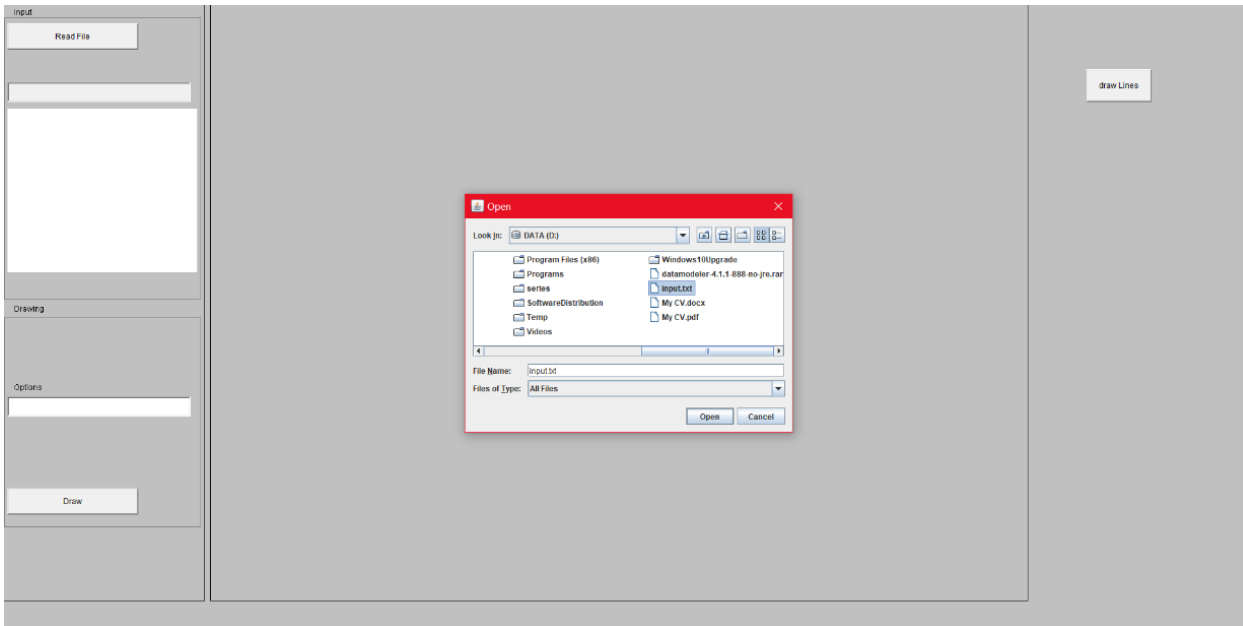
```
public void drawLines (Graphics g,int factor)
{
    int k=880*(factor);
    //to draw the black lines
    for(int i=100;i<880;i+=100)
    {
        g.setColor(Color.black);
        g.drawLine(330, i, 1500, i);
    }
    //to draw the dashed lines
    Graphics2D g2d = (Graphics2D) g;
    Stroke dashed = new BasicStroke(1, BasicStroke.CAP_BUTT, BasicStroke.JOIN_BEVEL, 0, new float[]{3}, 0);
    g2d.setStroke(dashed);

    for(int j=20;j<=900;j+=20)
    {
        g.drawLine(330,j, 1500, j);
        g.setFont(new Font("Arial",Font.PLAIN,15));
        g.drawString(String.valueOf(k), 1505, j);
        k-=20*(factor);
    }
}
```

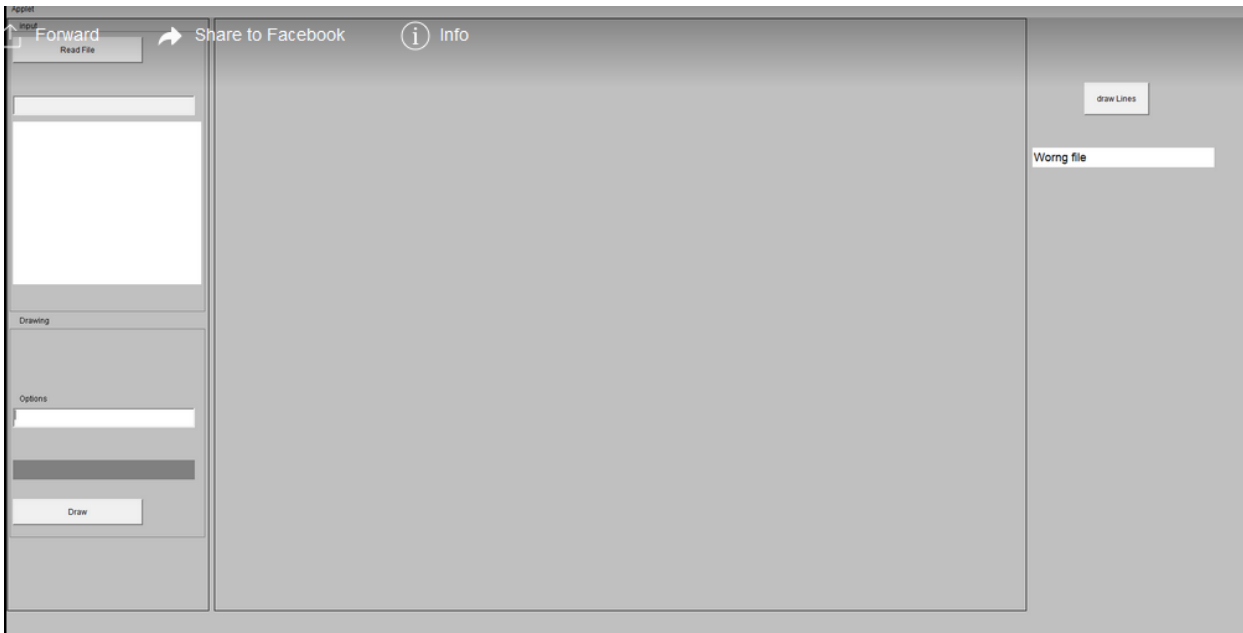

في ملف applet :
بعد صنع الواجهه وتزويدها بال buttons و labels و textFields
ووضعها في الأحداث المناسبة, ستظهر لدينا الواجهه التالية:



عند الضغط على الزر **ReadFile**: ستظهر لنا واجهة، نقوم من خلالها
باختيار الملف المراد.



يقوم البرنامج بالتأكد من أن لاحقة الملف المختار هو .txt. في حال لم تكن كذلك فإنه سيقوم بإظهار رسالة "wrong file"



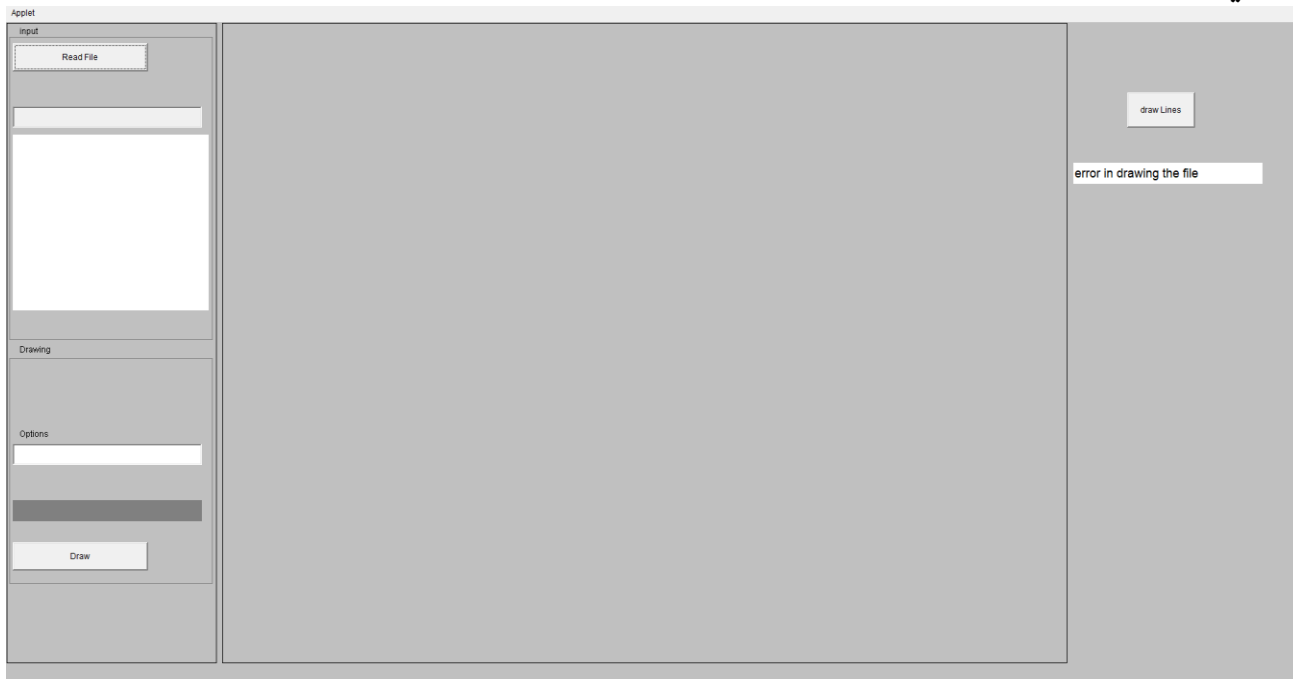
في حال كانت لاحقة .txt ولكن يحوي معلومات ناقصة أو خاطئة، فستظهر لدينا رسالة أنه لا يمكن رسم الملف.

مثال على ذلك:

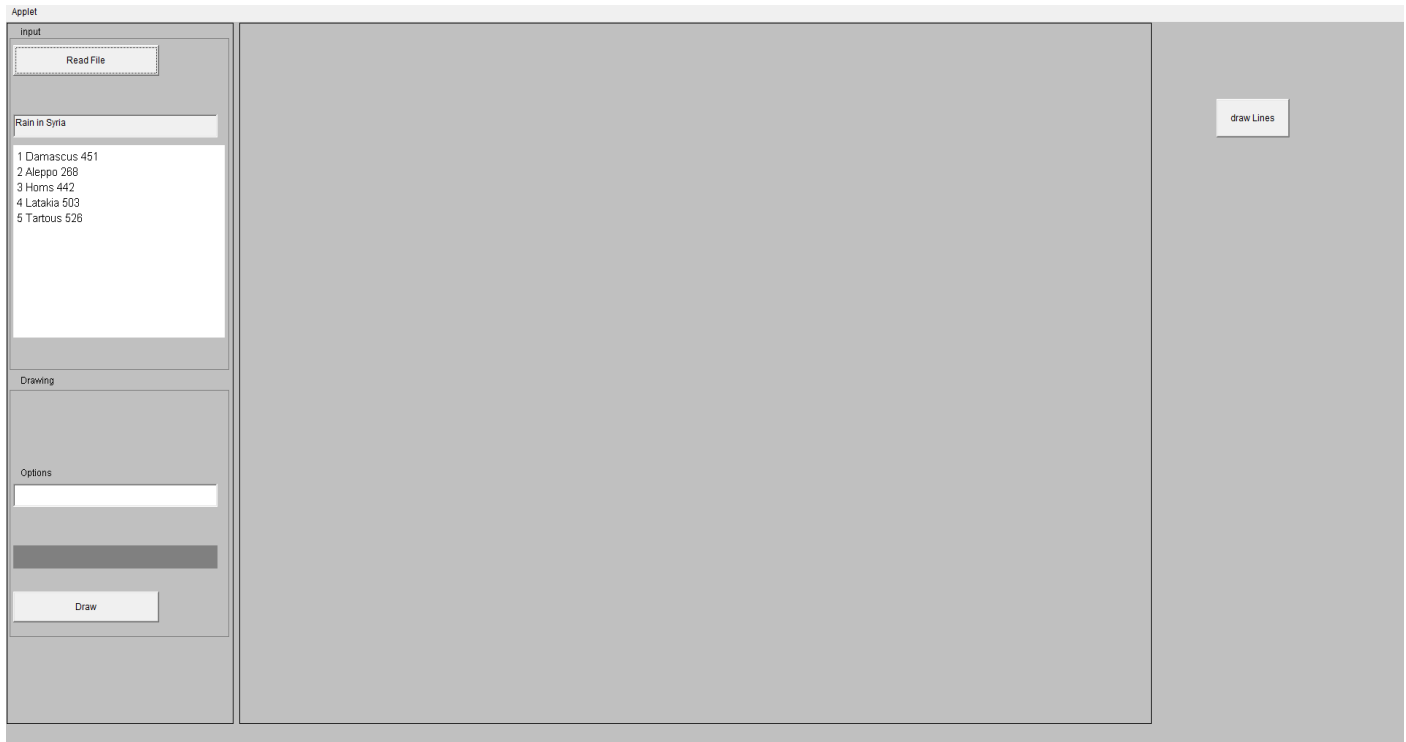
عند ادخال الملف input.txt الذي يحوي على المعلومات التالية:

```
File Edit Format View Help
Damascus
2007, 500
2008, 504
2009, 420
2010, 380
Aleppo
2007, 452
2008, 105
2009, 206
2010, 310
Homs
2007, 445
2008, 203
2009, 500
2010, 620
Latakia
2007, 500
2008, 641
2009, 751
2010, 123
Tartous
2007, 312
2008, 545
2009, 463
2010, 787
```

فمثلا في الملف السابق لا يوجد عنوان للملف، وبالتالي عند اختياره يظهر التالي:

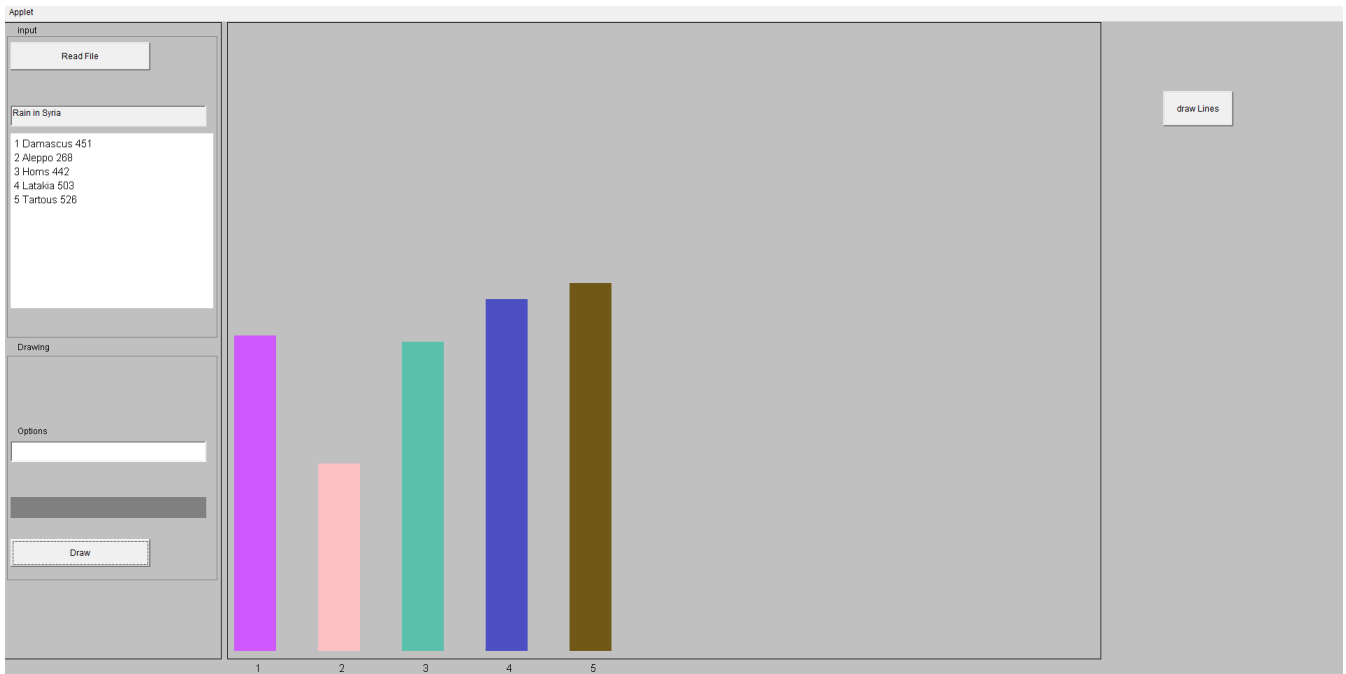


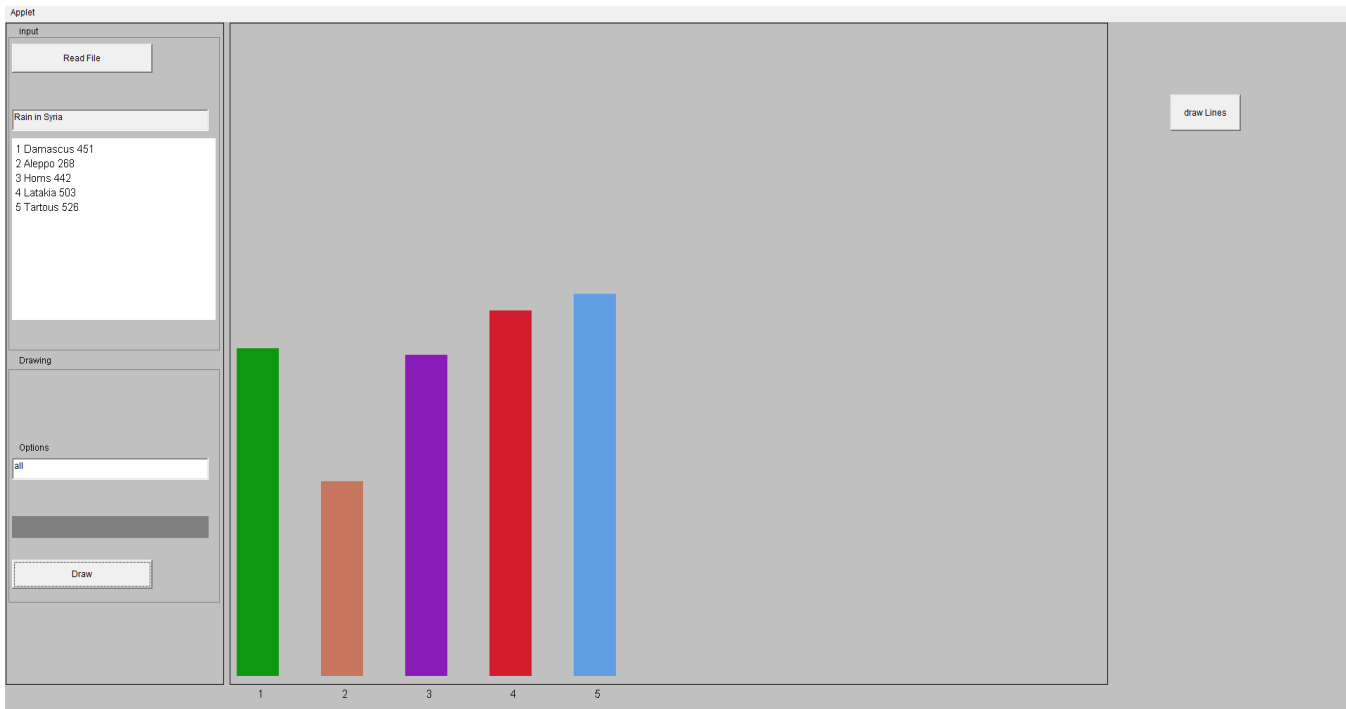
في حال كانت جميع مدخلات الملف صحيحة وفي صيغتها المناسبة ستظهر محتويات الملف في حقل موجود جانبا.



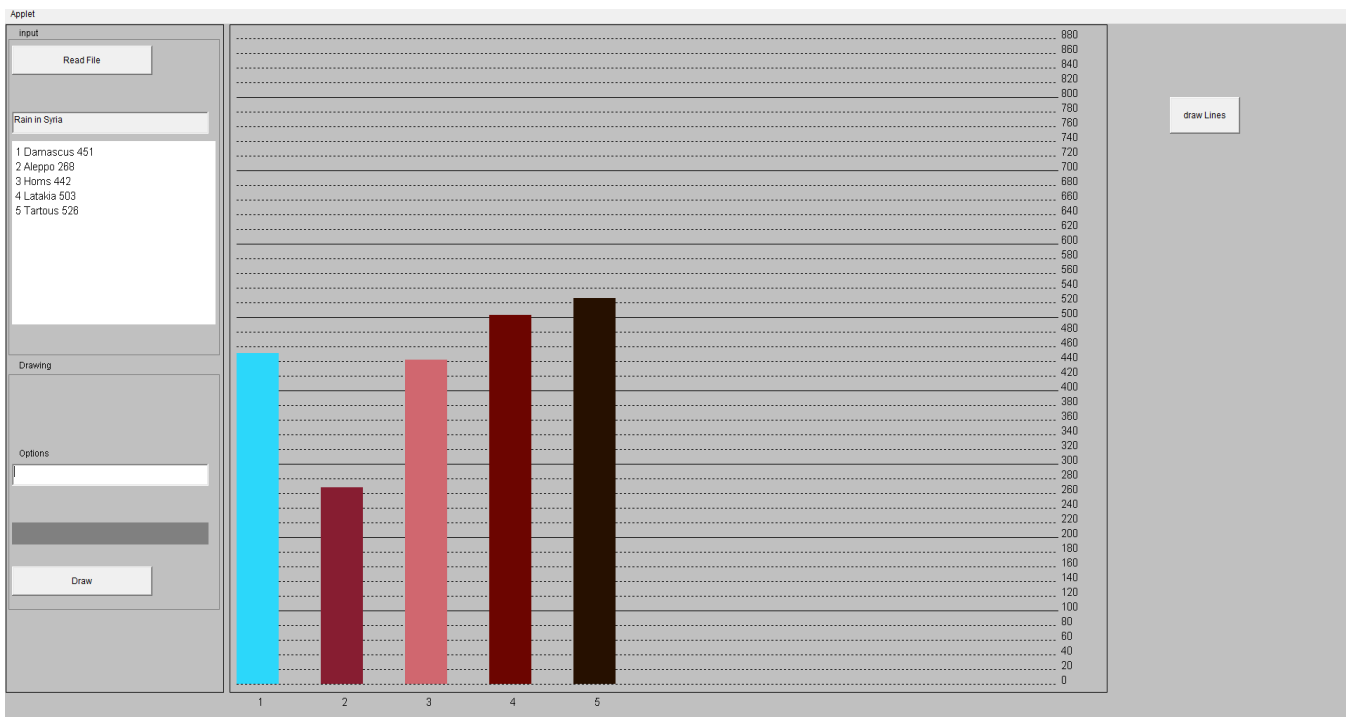
عند الضغط على زر **Draw**:

.....الحالة الأولى: حقل options فارغ أو كتابة كلمة all فيه ،سيقوم برسم معدلات هطول الأمطار في جميع المدن :

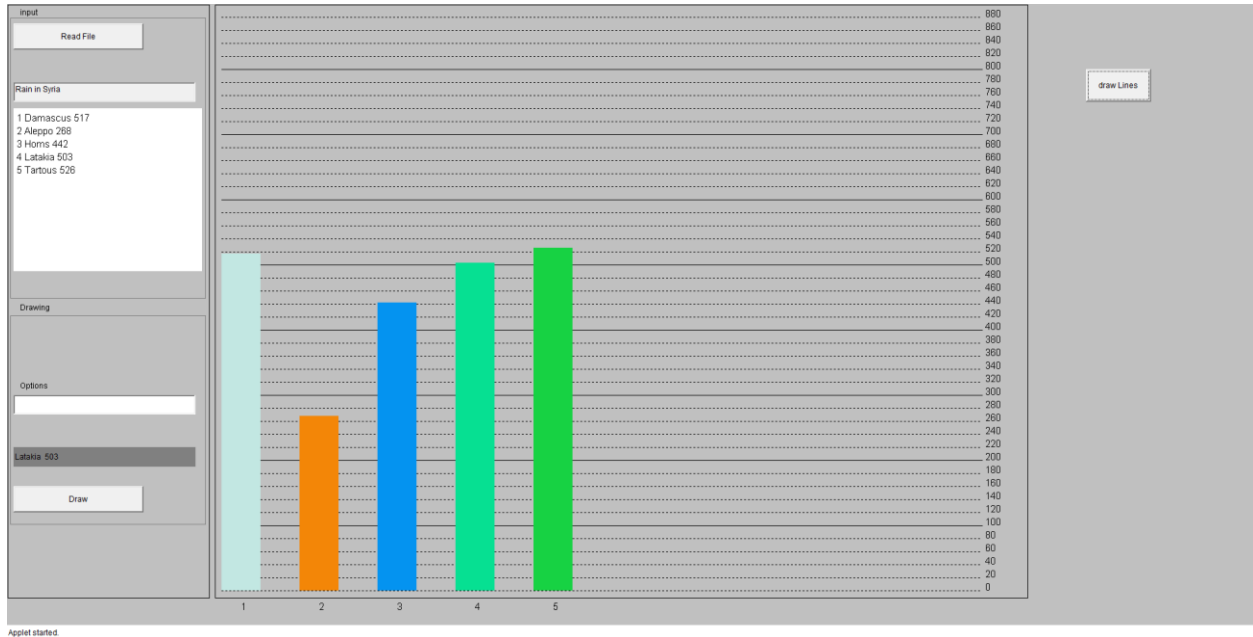




وعند الضغط مرة على زر **draw lines** سيقوم بإظهار الخطوط والتدرجات على الرسمة وسيقوم بإخفائها عند الضغط عليه مرة أخرى.

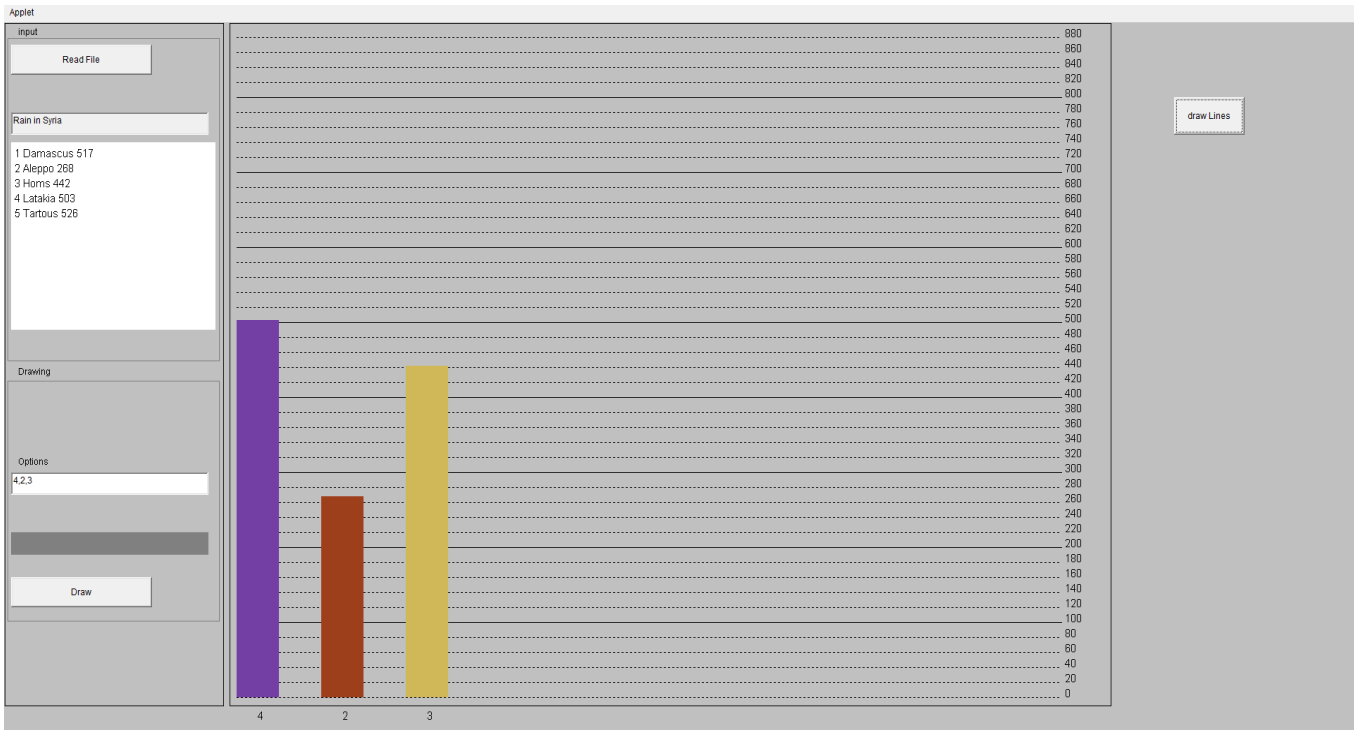


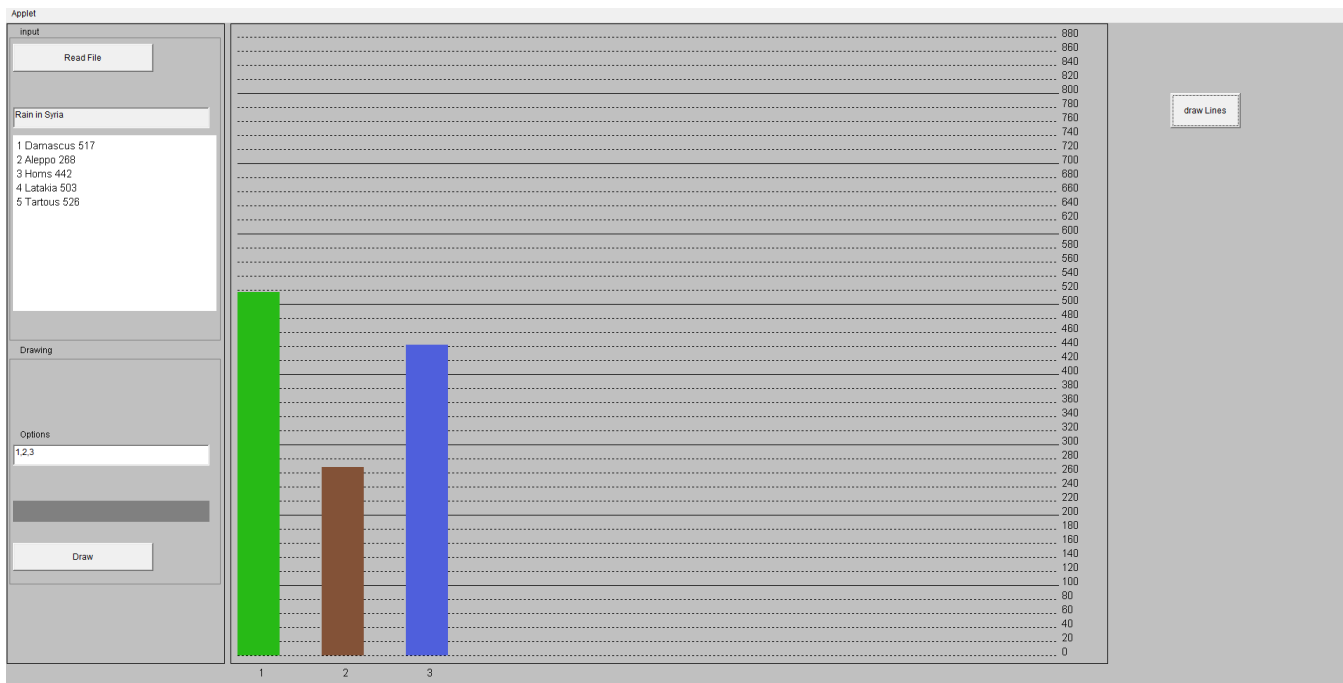
عند وضع الفأرة على أحد الأعمدة، سيظهر في حقل موجود جانباً اسم المدينة ومتوسط معدلات هطول الأمطار فيها.



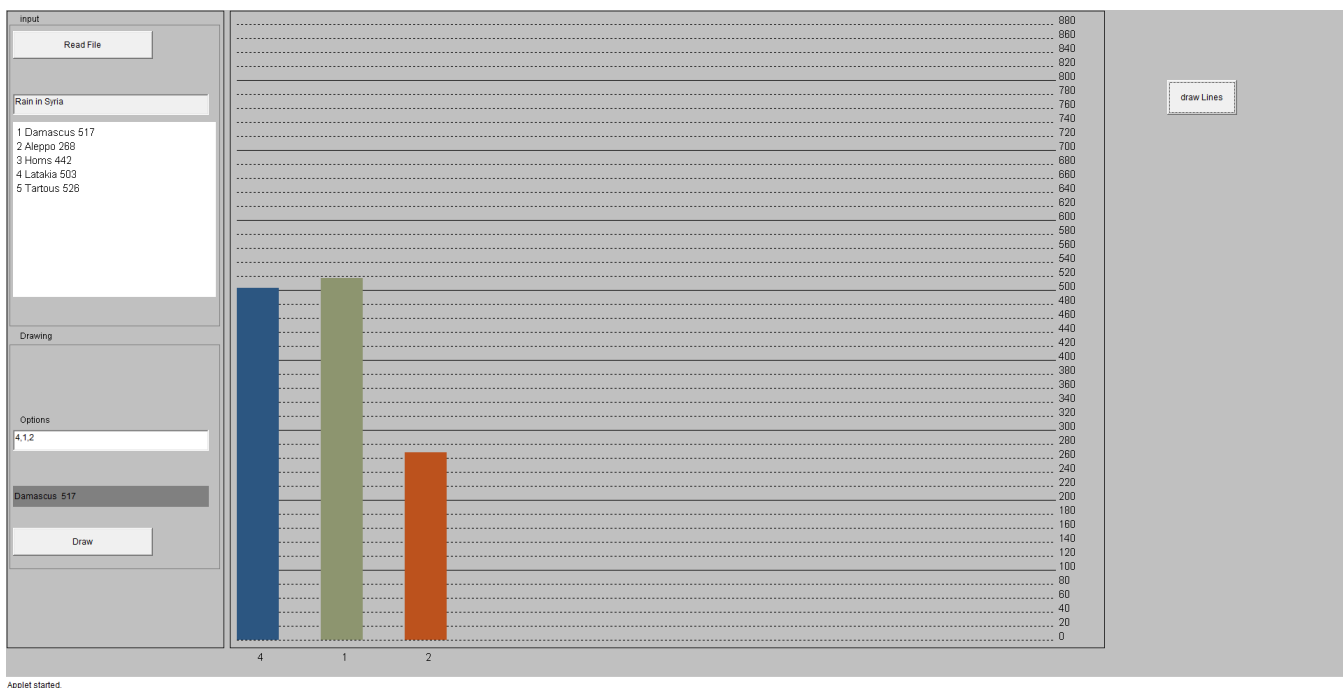
حيث أن مؤشر الفأرة موجود على العمود الرابع.

الحالة الثانية: إدخال مجموعة ارقام صحيحة تفصل بينها فواصل، سيقوم برسم متوسطات معدلات هطول الأمطار في هذه المدن.





وكما في الحالة الأولى، فإنه عند وضع المؤشر على أحد الأعمدة فسيظهر لنا في الحقل الموجود جانباً اسم المدينة ومتوسط معدلات هطول الأمطار فيها.



حيث أن المؤشر موجود على العمود الثاني.

في حال كان يوجد أي خطأ في الدخل كعدم وجود فواصل بين الأرقام أو وجود رقم اكبر من عدد المدن الموجودة في الملف أو إدخال محارف اخرى غير الفواصل ، ستظهر رسالة الخطأ "wrong options try agine"

Applet

input

Read File

Rain in Syria

1 Damascus 517
2 Aleppo 268
3 Horns 442
4 Latakia 503
5 Tartous 526

Drawing

Options

1 2 3

Draw

draw Lines

Wrong options try agine

Applet

input

Read File

Rain in Syria

1 Damascus 517
2 Aleppo 268
3 Horns 442
4 Latakia 503
5 Tartous 526

Drawing

Options

1 2

Draw

draw Lines

Wrong options try agine

Applet

input

Read File

Rain in Syria

1 Damascus 517
2 Aleppo 288
3 Homs 442
4 Latakia 503
5 Tartous 526

Drawing

Options

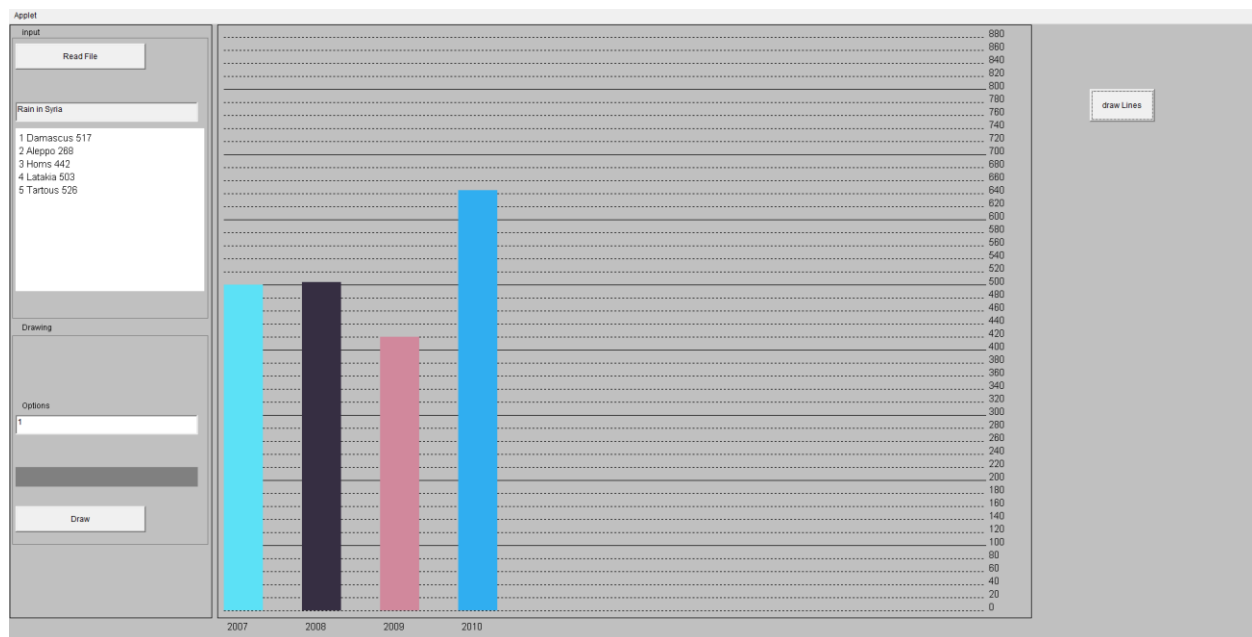
1.2.8

Draw

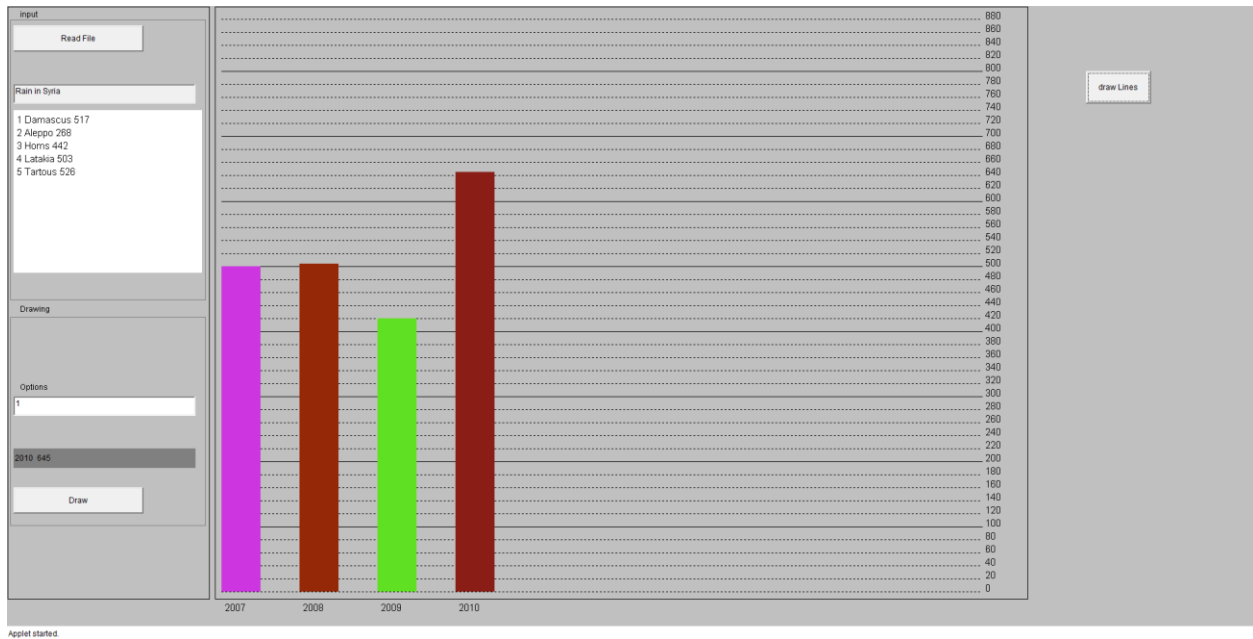
draw Lines

Wrong options try again

الحالة الثالثة: إدخال رقم وحيد قي حقل options، سيقوم برسم معدلات هطول الأمطار في كل سنة من سنوات هذه المدينة.

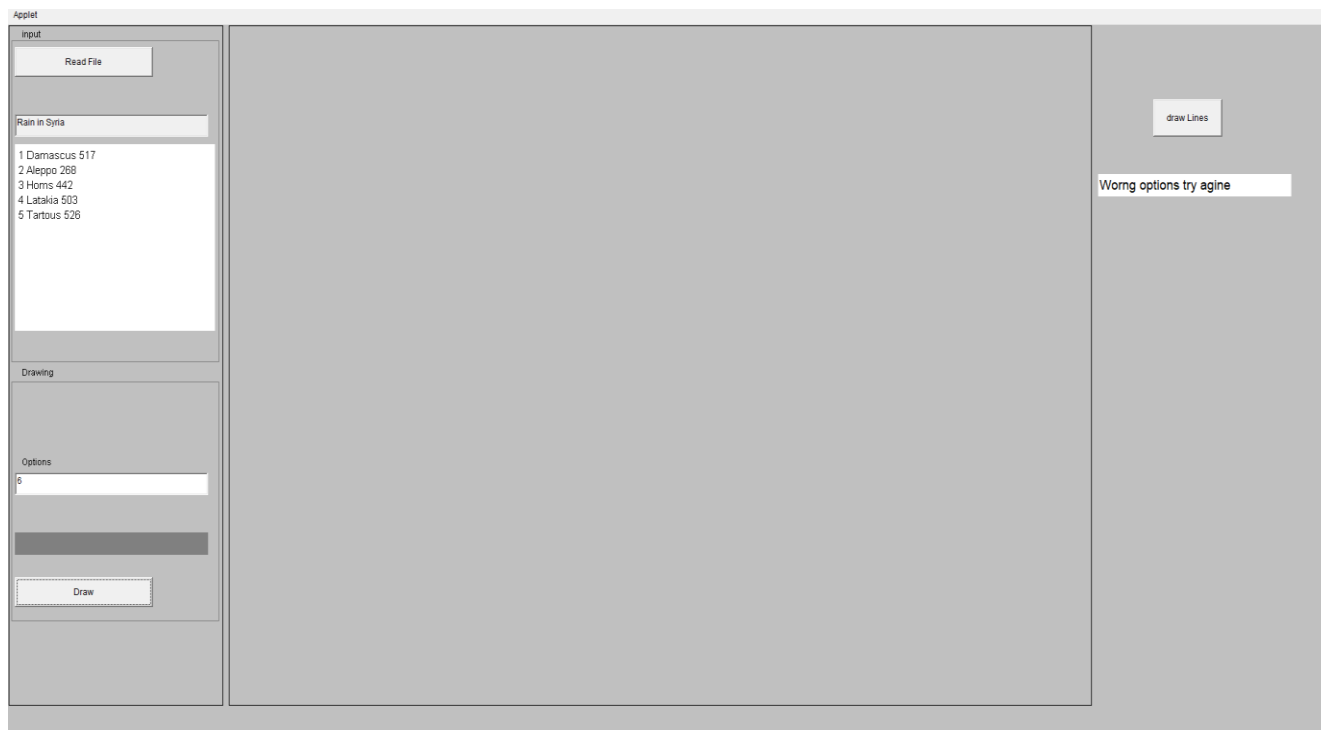


في هذه الحالة، عند وضع الفأرة على أحد الأعمدة، فإنه سيظهر لنا في الحقل جانباً العام ومعدل هطول الأمطار في هذا العام .



حيث أن مؤشر الفأرة موجود على العمود الرابع.

في حال إدخال رقم مدينة ليست موجودة في الملف ، ستظهر الرسالة " wrong option try agine".



في حال كان لدينا ملف يحوي المعلومات التالية:

input.txt - Notepad	input.txt - Notepad	input.txt - Notepad	input.txt - Notepad
File Edit Format View Help	File Edit Format View Help	File Edit Format View Help	File Edit Format View Help
Rain in Iceland	Keflavik	Hafnarfjörður	2024, 1425
Akureyri	2007, 2526	2007, 1453	2025, 3135
2007, 1000	2008, 1057	2008, 2035	2026, 2425
2008, 1046	2009, 1065	2009, 1004	Selfoss
2009, 2206	2010, 1104	2010, 2203	2007, 1234
2010, 2455	2011, 1213	2011, 1233	2008, 1370
2011, 1233	2012, 1000	2012, 960	2009, 850
2012, 2125	2013, 1325	2013, 1315	2010, 1350
2013, 1315	2014, 2135	2014, 1255	2011, 1642
2014, 1455	2015, 700	2015, 1752	2012, 1100
2015, 2452	2016, 900	2016, 1425	2013, 1000
2016, 1425	2017, 540	2017, 3455	2014, 978
2017, 3135	2018, 2425	2018, 2425	2015, 1234
2018, 2425	2019, 1000	2019, 1000	2016, 1475
2019, 1000	2020, 1700	2020, 1700	2017, 1520
2020, 1700	2021, 1534	2021, 1534	2018, 2425
2021, 1534	2022, 1872	2022, 1872	2019, 1000
2022, 1872	2023, 1304	2023, 1304	2020, 1700
2023, 1304	2024, 1425	2024, 1425	2021, 1534
2024, 1425	2025, 3135	2025, 3135	2022, 1872
2025, 3135	2026, 2425	2026, 2425	2023, 1304
2026, 2425	Hafnarfjörður	Húsavík	2024, 1425
Keflavik	2007, 1453	2007, 2005	2025, 3135
2007, 2526	2008, 2035	2008, 1412	2026, 2425

آلية الرسم:

باستخدام تابع `getFactorOfRates` نستطيع ايجاد مقياس رسم مناسب للأعمدة والتدرجات.

- من اجل رسم كل المدن:

قمنا بحساب متوسط معدلات هطول الامطار في كل مدينة
ومن اجل كل مدينة نقوم برسم العمود له باستخدام التعليمة :

```
g.fillRect(nums-30,900-(avgs[i-1]/factor),60,avgs[i-1]/factor);
```

حيث القيمة الابتدائية ل `nums` هي 360 ونقوم في كل مدينة بزيادة `nums` بمقدار 120 والدليل `i` يزداد من 1 الى عدد المدن الكلي.

حيث مبدأ احداثيات ال Histogram هو (330,900)

- من اجل رسم مدن اختيارية :

نستخدم مصفوفة `options` التي تحوي بداخلها الخيارات التي يريد المستخدم رسمها ومن اجل كل خيار نقوم برسم العمود له باستخدام التعليمة :

```
g.fillRect(nums-30 , 900-(avgs[cityindex-1])/factor , 60 ,  
(avgs[cityindex-1])/factor);
```

حيث `cityindex` هو دليل المدينة المختارة.

- من أجل رسم خيار واحد، نناقش حالتين:

الحالة الاولى : عدد السنوات اقل من 10

هنا يمكننا الرسم مثل الحالات السابقة فقط مع تبديل `avgs[]` بمصفوفة ال `Rates[]` التي تحوي معدلات هطول الامطار من اجل المدينة المختارة.

الحالة الثانية : عدد السنوات اكبر من 10

نستخدم المتحول v الذي سوف يساعدنا بمعرفة عرض كل عمود والمسافة الفاصلة بين الاعمدة.

```
int v=(1200/len)-5;
```

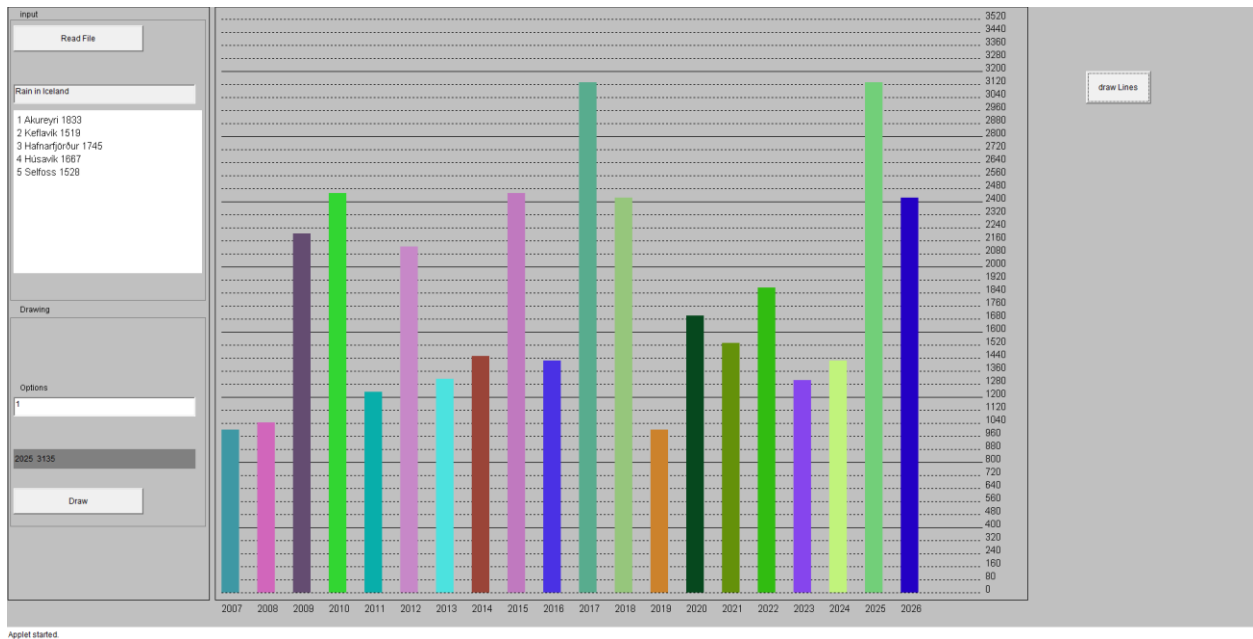
حيث len هو عدد السنوات الكلي.

ويكون $nums$ في هذه الحالة هو $nums=330+(v/2)$

ونرسم الاعمدة لكل سنة عن طريق التعليمة :

```
g.fillRect(nums-(v/2),900-Rates[i]/factor), (v/2), Rates[i]/factor);
```

ومن اجل كل سنة نقوم بزيادة $nums$ بمقدار v .



حيث أن مؤشر الفأرة على العمود قبل الأخير (سنة 2025)