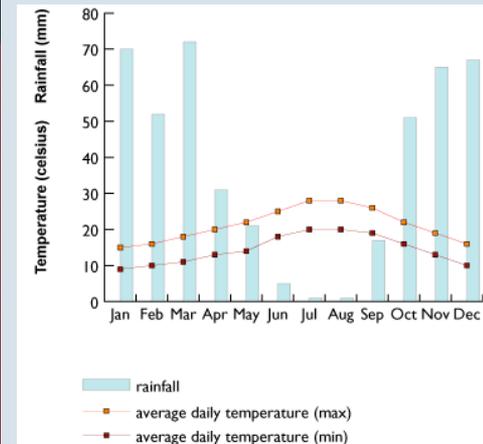
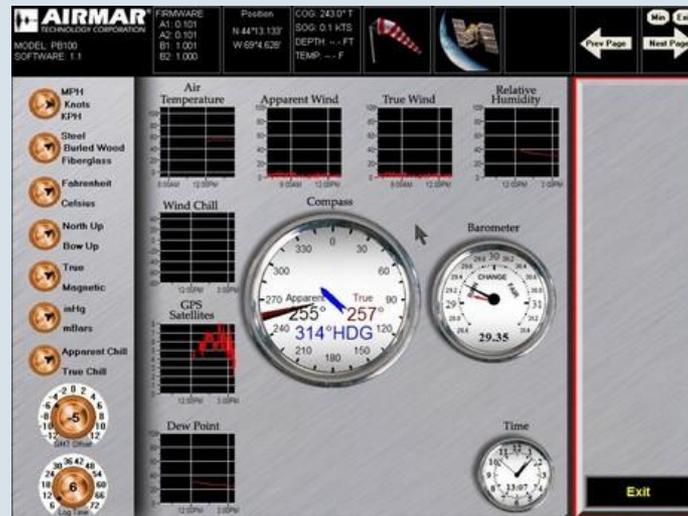


Leitura e Escrita em Ficheiros

Estação Meteorológica

Estação Meteorológica

- Defina em Java uma classe `WeatherStation` cujos objectos representam uma estação meteorológica.
- Programe a sua classe no Eclipse.
- Teste um (ou vários) objectos `WeatherStation`, e verifique se se comportam como se espera.



Estação Meteorológica

- Cada objecto `WeatherStation` recebe valores de temperatura ao longo do tempo e guarda o máximo, o mínimo e a média das temperaturas registadas
- Operações reconhecidas

```
public void sampleTemperature(double temp)
```

Registar a amostra `temp` na estação

```
public double getMaximum()
```

Consultar a máxima temperatura observada até ao momento

```
public double getMinimum()
```

Consultar a mínima temperatura observada até ao momento

```
public double getAverage()
```

Consultar a média das temperaturas observadas até ao momento

```
public void reset(double temp)
```

Apagar a informação registada na estação, para se iniciarem novas medições. É indicada também uma primeira nova medição.

Estação Meteorológica

- Vamos desenvolver o programa principal da classe `WeatherStation`
 - O programa deve receber várias medições de temperatura (`double`)
 - O programa deve terminar quando receber um valor não numérico
- Para conseguir isto é necessário fazer algumas modificações na classe
 - Vamos criar um construtor sem parâmetros para a classe `WeatherStation`
 - Vamos modificar o método `sampleTemperature()`, de forma a que este possa receber todas as medições de temperatura, incluindo a primeira
- Na classe `Main` (programa principal):
 - Na condição de manutenção do ciclo vamos usar o método `hasNextDouble()` da classe `Scanner`.

Estação Meteorológica

```
// Novo construtor
public WeatherStation() {
    min = 0;
    max = 0;
    counter = 0;
    accum = 0;
}

//Registrar a amostra temp na estação
public void sampleTemperature(double temp) {
    if (counter == 0) {
        min = temp;
        max = temp;
    } else {
        if (temp > max)
            max = temp;
        if (temp < min)
            min = temp;
    }
    counter++;
    accum += temp;
}
```

Modificações a inserir na
classe WeatherStation

Estação Meteorológica – Classe `Main`

```
import java.util.Scanner;
```

```
public class Main{  
    public static void main(String[] args) {  
        WeatherStation ws;  
        Scanner in;  
        double temp;  
        in=new Scanner(System.in);  
        ws = new WeatherStation();  
        System.out.print("Temperatura (caracter não numérico para fim): ");  
        while (in.hasNextDouble()) {  
            temp=in.nextDouble();  
            in.nextLine();  
            ws.sampleTemperature(temp);  
            System.out.print("Temperatura (caracter não numérico para fim): ");  
        }  
        System.out.println("Estatísticas");  
        System.out.println("Temperatura máxima: " + ws.getMaximum());  
        System.out.println("Temperatura mínima: " + ws.getMinimum());  
        System.out.printf("Média: %.2f\n", ws.getAverage());  
    }  
}
```

O ciclo vai decorrer enquanto o utilizador inserir valores que são aceites como `double`.

Cada valor inserido é adicionado à estação

Depois de terminada a leitura, apresentam-se os resultados

E se os dados estivessem
num ficheiro ?

Leitura de ficheiros

Escrita em ficheiros

Ler ficheiros de texto

- A forma mais simples de ler texto é usar a classe `Scanner`
- Para ler de um ficheiro do disco, contrói-se um objecto da classe `FileReader` (do pacote `java.io`)
- Depois, usa-se o `FileReader` para construir um objecto `Scanner`

```
FileReader reader = new FileReader("input.txt");  
Scanner in = new Scanner(reader);
```

- Utilizam-se os métodos de `Scanner` para ler a informação
 - `next`, `nextLine`, `nextInt`, e `nextDouble`

Escrever ficheiros de texto

- Para escrever num ficheiro, constrói-se um objecto da classe `PrintWriter` (também `java.io`)

```
PrintWriter out = new PrintWriter("output.txt");
```

- Se o ficheiro já existe, o seu conteúdo é apagado antes das novas acções de escrita
- Se o ficheiro não existir, é criado um ficheiro vazio

Escrever ficheiros de texto

- Usam-se `print` e `println` para escrever num objecto `PrintWriter`:

```
out.println(29.95);  
out.println(new Rectangle(5, 10, 15, 25));  
out.println("Hello, World!");
```

- Terminado o processamento do ficheiro, este deve ser fechado:

```
out.close();
```

- Senão, o output pode não ser todo escrito no ficheiro

Classe Main – com ficheiros

```
import java.util.Scanner;
import java.io.*;
public class Main {
    public static void main(String[] args) throws Exception {
        WeatherStation ws;
        Scanner in;
        double temp;
        FileReader reader = new FileReader("sample.txt");
        PrintWriter out = new PrintWriter("statistics.txt");
        in = new Scanner(reader);
        ws = new WeatherStation();
        while (in.hasNextDouble()) {
            temp=in.nextDouble();
            in.nextLine();
            ws.sampleTemperature(temp);
        }
        out.println("Estatísticas");
        out.println("Temperatura máxima: "+ws.getMaximum());
        out.println("Temperatura mínima: "+ws.getMinimum());
        out.printf("Média: %.2f\n",ws.getAverage());
        out.close();
    }
}
```

E se não for
introduzido nenhum
valor ?

Classe Main – com ficheiros

```
import java.util.Scanner;
import java.io.*;
public class Main {
    public static void main(String[] args) throws Exception {
        WeatherStation ws;
        Scanner in;
        double temp;
        FileReader reader = new FileReader("sample.txt");
        PrintWriter out = new PrintWriter("statistics.txt");
        in = new Scanner(reader);
        ws = new WeatherStation();
        while (in.hasNextDouble()) {
            temp=in.nextDouble();
            in.nextLine();
            ws.sampleTemperature(temp);
        }
        if (ws.getNumberTemperatures() > 0) {
            out.println("Estatisticas");
            out.println("Temperatura maxima: "+ws.getMaximum());
            out.println("Temperatura mínima: "+ws.getMinimum());
            out.printf("Média: %.2f\n",ws.getAverage());
        }
        else out.println("Não foram introduzidas temperaturas. ");
        out.close();
    }
}
```

Seria preciso implementar o método
`getNumberTemperatures()`