

Bài 1. BÀI TẬP VỀ CẤU TRÚC ĐIỀU KHIỂN VÀ MẢNG

1. Tạo 1 Project: Phan1

2. Tạo lớp BtMang bao gồm các hàm sau:

a. Hàm TaoNgay(String ngay1, String ngay2): Hàm trả về 1 ngày ngẫu nhiên từ ngay1 đến ngay2.

```
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.Random;

public class BtMang {
    public String TaoNgay(String ngay1, String ngay2){
        try{

            //Định dạng ngày theo dạng ngày tháng năm
            SimpleDateFormat f= new SimpleDateFormat("dd/MM/yyyy");
            Date n1= f.parse(ngay1); //Đổi chuỗi ngày1 ra kiểu ngày
            Date n2= f.parse(ngay2);
            Random r= new Random(); //Kích hoạt để lấy số ngẫu nhiên
            while(true){
                Long t=r.nextLong(); //Lấy 1 số ngẫu nhiên
                //n1.getTime: Đổi ngày ra số
                if(t>=n1.getTime()&&t<=n2.getTime())
                {
                    Date n= new Date(t); //Đổi số ra ngày
                    return f.format(n); //Trả về chuỗi ngày tháng năm
                }
            }
        }catch(Exception tt){
            System.out.println("Lỗi: "+ tt.getMessage());
            return null;
        }
    }
}
```

Chú ý: Tạo thêm lớp Mang bao gồm hàm main để gọi lại hàm TaoNgay của lớp BtMang:

```
public class Mang {

    public static void main(String[] args) {
        BtMang m= new BtMang();
        String ngay1="01/02/1990";
        String ngay2="12/12/1992 ";
        System.out.println(m.TaoNgay(ngay1,ngay2));
    }
}
```

Chạy chương trình để kiểm tra

b. Mở lớp BtMang và tạo hàm TaoHoTen để trả về một họ tên ngẫu nhiên

```

public String TaoHoTen(){
    String[] ho={"Tran", "Le","Nguyen"};
    String[] chulot={"Thanh", "Hoang","Trung", "Van","My"};
    String[] ten={"Hung", "Nga","Tien","Tam","Ton","Giang","Sy"};
    Random r= new Random();
    //ho.length tra ve do dai cua mang ho ->3
    String h=ho[r.nextInt(ho.length)];
    String cl=chulot[r.nextInt(chulot.length)];
    String t=ten[r.nextInt(ten.length)];
    return h+" "+cl+" "+t;
}

```

Chú ý: Mở lớp Mang để gọi lại hàm TaoHoTen của lớp BtMang, ngày sinh được tạo ngẫu nhiên từ ngày 01/02/1990 đến 12/12/2000:

```

public class Mang {

    public static void main(String[] args) {
        BtMang m= new BtMang();
        System.out.println(m.TaoHoTen());
        String ngay1="01/02/1990";
        String ngay2="12/12/1992 ";
        System.out.println(m.TaoNgay(ngay1,ngay2));
    }
}

```

Yêu cầu bổ sung: *Hiển thị ra 100 họ tên và ngày sinh một cách ngẫu nhiên*

c. Mở lớp BtMang và tạo hàm HienThi(int n) để hiển thị họ tên, ngày sinh, điểm trung bình của n sinh viên:

```

public void HienThi(int n){
    Random r= new Random();
    for(int i=1;i<=n;i++){
        double dtb= r.nextDouble()*10;
        System.out.println(TaoHoTen()+" "+TaoNgay("01/01/1960","01/01/2000")+"; "+dtb);
    }
}

```

3. Yêu cầu bổ sung:

- Mở lớp Mang để:
 - Hiển thị thông tin của 10 sinh viên.
 - Hiển thị họ tên, ngày sinh, giới tính, điểm trung bình, kết quả và xếp loại của các sinh viên. Trong đó:
 - Kết quả = đậu nên điểm trung bình ≥ 5 , ngược lại là rớt
 - Giới tính: Nam hoặc nữ lấy ngẫu nhiên
 - Xếp loại học viên theo điểm trung bình (loại giỏi từ 8.0 trở lên, khá từ 6.5 đến 7.9, trung bình từ 5 đến 6.4, yếu từ 3.1 đến 4.9, kén dưới 3)
 - Hiển thị n sinh viên (n nhập từ bàn phím)

Bài 2. BÀI TẬP VỀ FILE

1. Mở lại Project Phan1 và tạo lớp BtFile bao gồm các hàm:

- a. Hàm TaoDanhSach(int n) dùng để tạo ra thông tin của n sinh viên, các thông tin này được lấy ngẫu nhiên. Lưu n sinh viên này vào file sv.txt ở thư mục Phan1.

```
import java.io.FileOutputStream;
import java.io.OutputStream;
import java.io.OutputStreamWriter;
import java.io.PrintWriter;
import java.util.Random;

public class BtFile {
    public void TaoDanhSach(int n)
    {
        try{
            //Tạo file sv.txt lưu ở thư mục Phan1
            FileOutputStream f= new FileOutputStream ("sv.txt");
            OutputStreamWriter o= new OutputStreamWriter(f);
            PrintWriter w= new PrintWriter(o);

            Random r= new Random();
            //tạo lớp BtMang để sử dụng các hàm bên trong lớp này
            BtMang m= new BtMang();
            for(int i=1;i<=n;i++){//Duyệt qua n sinh viên
                double dtb= r.nextDouble()*10;//Tạo ngẫu nhiên 1 đtb
                //Ghi vào file mã sv, họ tên, ngày sinh và điểm trung bình
                w.println("Sv"+i+";"+m.TaoHoTen()+";"+m.TaoNgay("01/01/1960","01/01/2000")+";"+dtb);
            }
            //Đóng file và lưu dữ liệu
            w.close();
            System.out.println("Đã tạo xong");
        }catch(Exception tt){
            System.out.println("Lỗi o hàm TaoDanh sach:"+ tt.getMessage());
        }
    }
}
```

- b. Hàm XuatDanhSach() dùng để xuất thông tin của các sinh viên từ file sv.txt ra màn hình.

```
public void XuatDanhSach()
{
    try{
        //Mở file sv.txt lưu ở thư mục Phan1
        FileInputStream f= new FileInputStream ("sv.txt");
        InputStreamReader ir= new InputStreamReader(f);
        BufferedReader r= new BufferedReader(ir);
        while(true){//Duyệt file
            String st=r.readLine();
            if(st=="||st==null) break;//Nếu hết file: hoặc st="" hoặc st=null
            String[] ds=st.split(";");//Tách các thông tin từ st
            System.out.println(ds[0]+"\\n"+ds[1]+"\\n"+ds[2]+"\\n"+ds[3]+"\\n\\n");
        }
        //Đóng file
        r.close();
    }catch(Exception tt){
        System.out.println("Lỗi o hàm XuatDanh sach:"+ tt.getMessage());
    }
}
```

2. Yêu cầu bổ sung:

- Hiện thị họ tên, ngày sinh, giới tính, điểm trung bình, kết quả và xếp loại của các sinh viên. Trong đó:

- Kết quả = đậu nên điểm trung bình ≥ 5 , ngược lại là rớt
- Giới tính: Nam hoặc nữ lấy ngẫu nhiên
- Xếp loại học viên theo điểm trung bình (loại giỏi từ 8.0 trở lên, khá từ 6.5 đến 7.9, trung bình từ 5 đến 6.4, yếu từ 3.1 đến 4.9, kén dưới 3)
- Viết thêm các hàm để:
 - Tìm kiếm (tương đối) theo tên của sinh viên
 - Đếm số sinh viên đậu, rớt
 - Tính trung bình cộng của điểm trung bình
 - Mở file sv.txt và in ra các dòng không hợp lệ:
 - Ngày sinh phải định dạng theo dd/MM/yyyy và có đủ 10 ký tự
 - Điểm trung bình phải từ 0..10
 - Chú ý: mở file sv.txt sửa lại dữ liệu để kiểm tra
- Tạo thêm 1 lớp chứa hàm main để gọi lại các hàm đã tạo.

Bài 3. BÀI TẬP VỀ ARRAYLIST

1. Mở lại Project Phan1 và tạo lớp BtArraylist bao gồm các hàm:

- a. **TaoDanhSach():** Hàm đọc file sv.txt và lưu họ tên, điểm trung bình vào 2 mảng động
- b. **XuatDanhSach():** Hàm xuất họ tên và điểm trung bình từ mảng động

```
public class BtArraylist {  
    //Tạo mảng động để lưu các họ tên sinh viên  
    public ArrayList<String> DsHoten= new ArrayList<String>();  
    //Tạo mảng động để lưu các dtb của sinh viên  
    public ArrayList<Double> DsDtb= new ArrayList<Double>();  
    public void TaoDanhSach(){  
        try{  
            //Mở file sv.txt lưu ở thư mục Phan1  
            FileInputStream f= new FileInputStream ("sv.txt");  
            InputStreamReader ir= new InputStreamReader(f);  
            BufferedReader r= new BufferedReader(ir);  
            while(true){//Duyệt file  
                String st=r.readLine();  
                if(st=="||st==null) break;//Nếu hết file: hoặc st="" hoặc st=null  
                String[] ds=st.split(";");//Tách các thông tin từ st  
                DsHoten.add(ds[1]);//Thêm vào mảng DsHoten 1 họ tên  
                //Thêm vào mảng DsDtb 1 điểm trung bình  
                DsDtb.add(Double.parseDouble(ds[3]));  
            }  
            //Đóng file  
            r.close();  
        }catch(Exception tt){  
            System.out.println("Loi o ham XuatDanh sach:"+ tt.getMessage());  
        }  
    }  
    public void XuatDanhSach(){  
        int ss=DsHoten.size();//Lấy độ dài của mảng  
        for(int i=0;i<ss;i++){//Duyệt qua các phần tử của mảng  
            System.out.println(DsHoten.get(i)+":"+ DsDtb.get(i));  
        }  
    }  
}
```

2. **Yêu cầu bổ sung:**

- Viết thêm các hàm để thao tác trên DsHoTen và DsDtb:
 - o Tìm kiếm (tương đối) theo tên của sinh viên.
 - o Thống kê số sinh viên đậu, rớt
 - o Tính trung bình cộng của điểm trung bình
- Tạo thêm 1 lớp chứa hàm main để gọi lại các hàm đã tạo.

Bài 4. Bài tập tự giải.

Sinh viên tự tạo ra 1 Project mới để thực hành

Mô tả chung về bài toán:

Cho một hệ thống quản lý bãi đỗ xe của khu chung cư được mô tả như sau:

- Xe vào gửi là xe của cư dân sống tại chung cư nên thông tin của mỗi xe phải bao gồm: Biển số xe (xe đạp không có biển số), số CMND của chủ nhân, họ tên, giới tính, mã số căn hộ, hãng sản xuất (Xe đạp thì không bắt buộc phải có hãng sản xuất), màu sơn.
- Xe cần quản lý ở bãi được chia thành thành 2 loại: xe 4 bánh (xe ô tô) và xe 2 bánh.
- Xe 2 bánh được chia thành 2 loại: xe gắn máy và xe đạp.
- Đối với xe 4 bánh thì khi đỗ xe hoặc lấy xe, nhân viên quản lý bãi đỗ xe phải ghi nhận các thông tin: **biển số xe (biển kiểm soát), thời gian vào bãi, thời gian ra bãi, tình trạng xe lúc vào bãi, tình trạng xe lúc ra bãi.**
- Đối với xe gắn máy, nhân viên quản lý cần ghi nhận các thông tin: **biển số xe, thời gian vào bãi, thời gian ra bãi.**
- Đối với xe đạp, nhân viên quản lý cần ghi nhận các thông tin: **số vé xe (do nhân viên quản lý bãi tự cấp cho chủ xe khi giao vé giữ xe), thời gian vào bãi, thời gian ra bãi.**

Yêu cầu:

Xây dựng chương trình viết bằng ngôn ngữ lập trình Java đáp ứng các yêu cầu sau:

(Lưu ý: không cần xử lý tiếng Việt có dấu khi đọc/ghi file dữ liệu)

Câu 1 (1 điểm): Đọc được file dữ liệu đầu vào cho chương trình có tên là **input1.txt**, cấu trúc của file này được mô tả như sau:

Mỗi dòng của file tương ứng với dữ liệu của một xe **đi vào bãi đỗ xe**. Dữ liệu được mô tả theo thứ tự như sau:

Loại xe, biển số xe, số vé xe, thời gian vào bãi, tình trạng xe lúc vào bãi. Tùy vào từng loại xe mà một số thông tin sẽ không có giá trị. Những thông tin nào không có thì sẽ nhận giá trị là "Not Available". Các thông tin trên được phân cách với nhau bởi dấu chấm phẩy (;).

Loại xe được quy định như sau: 4 là xe 4 bánh, 2 là xe gắn máy, 0 là xe đạp.

Tình trạng xe lúc vào bãi được quy định như sau: Nếu xe không có bất thường nào thì có giá trị là "Bình thường". Nếu xe có bất kỳ bất thường nào (ví dụ bị trầy xước, mất kính chiếu hậu...) thì phải mô tả rõ.

Ví dụ mẫu về dữ liệu của file **input1.txt** được mô tả như hình sau:

```
4;37A-365.12;Not Available;11:30 07/11/2015;Bình thường
0;Not Available;1014;14:15 07/11/2015;Not Available
4;37A-534.74;Not Available;13:05 09/12/2015;Tray xuat ben cua phia truoc
2;38H1-4536;Not Available;17:10 17/11/2015;Not Available
```

Câu 2 (1 điểm): Đọc được file dữ liệu đầu vào thứ hai cho chương trình có tên là **input2.txt**, cấu trúc của file này được mô tả như sau:

Mỗi dòng của file tương ứng với dữ liệu của một xe **đi ra khỏi bãi đỗ xe**. Dữ liệu được mô tả theo thứ tự như sau:

Loại xe, biển số xe, số vé xe, thời gian ra bãi, tình trạng xe lúc ra bãi. Tùy vào từng loại xe mà một số thông tin sẽ không có giá trị. Những thông tin nào không có thì sẽ nhận giá trị là "Not Available". Các thông tin trên được phân cách với nhau bởi dấu chấm phẩy (;).

Loại xe được quy định như sau: 4 là xe 4 bánh, 2 là xe gắn máy, 0 là xe đạp.

Giá trị của thông tin "Tình trạng xe lúc ra bãi" (dành cho xe 4 bánh) được mô tả như sau:

- Nếu xe không có phát sinh bất thường nào khi ra bãi (so với lúc vào bãi) thì giá trị của thông tin này là "Bình thường".

- Nếu xe có phát sinh bất thường nào khi ra bãi (so với lúc vào bãi) thì giá trị của thông tin này là những mô tả về các bất thường đó.

Ví dụ mẫu về dữ liệu của file input2.txt được mô tả như hình sau:

```
4;37A-365.12;Not Available;15:30 07/11/2015;Bình thường
0;Not Available;1014;15:00 07/11/2015;Not Available
4;37A-534.74;Not Available;7:05 10/12/2015;Bi hong kinh chieu hau
2;38H1-4536;Not Available;8:10 20/11/2015;Not Available
```

Câu 3 (4 điểm): Xuất thông tin ngày giờ vào/ra và tính tiền thuê bãi cho các xe ra file output.txt:

Lưu ý:

- **Giả định thông tin trong file input2.txt là những thông tin đã hợp lệ. Thí sinh không cần kiểm tra tính hợp lệ của dữ liệu trong file input2.txt.** (Ví dụ: trong file input2.txt có dữ liệu liên quan đến 1 xe 4 bánh có biển số xe là 75A-571.47 nghĩa là giả định trong file đã chắc chắn có dữ liệu của xe 4 bánh có biển số xe như vậy rồi)

- **Cách tính tiền thuê bãi như sau:**

+ **Đối với xe 4 bánh: 5000 đồng / 0,5 giờ** (tức là làm tròn đến 0,5 giờ - ví dụ thuê 1 giờ 5 phút thì được tính thành 1,5 giờ; thuê 35 phút thì làm tròn thành 1 giờ; thuê 5 phút cũng được làm tròn thành 0,5 giờ). **Riêng trường hợp có phát sinh tình trạng bất thường của xe khi ra bãi thì giá trị của thông tin tiền thuê bãi này được ghi là: "Chưa xác định - Đang xử lý bồi thường".**

+ **Đối với xe gắn máy: 3000 đồng / ngày đêm** (tức là làm tròn đến 1 ngày - ví dụ thuê 1 ngày 1 phút thì được tính thành 2 ngày; thuê 5 phút cũng được làm tròn thành 1 ngày)

+ **Đối với xe đạp: 1000 đồng / ngày đêm** (tức là làm tròn đến 1 ngày - ví dụ thuê 1 ngày 1 phút thì được tính thành 2 ngày; thuê 5 phút cũng được làm tròn thành 1 ngày)

Chú ý: Cách xuất thông tin ra file output.txt được mô tả trong ví dụ ở hình sau:

```
4;37A-365.12;Not Available;15:30 07/11/2015;17:00 07/11/2015;15000
0;Not Available;1014;15:00 07/11/2015;19:00 07/11/2015;1000
4;37A-534.74;Not Available;7:05 10/12/2015;17:45 10/12/2015;Chưa xác định - Đang xử lý bồi thường
2;38H1-4536;Not Available;8:10 20/11/2015;6:00 21/11/2015;3000
```

Mỗi dòng của file tương ứng với dữ liệu của một xe đã vào và ra bãi đỗ xe. Dữ liệu được mô tả theo thứ tự như sau:

Loại xe, biển số xe, số vé xe, thời gian vào bãi, thời gian ra bãi, tiền thuê bãi. Tùy vào từng loại xe mà một số thông tin sẽ không có giá trị. Những thông tin nào không có thì sẽ nhận giá trị là "Not Available". Các thông tin trên được phân cách với nhau bởi dấu chấm phẩy (;).