

Sau khi thực hành xong bài này, học viên có khả năng :

- Sử dụng được generic cho tập hợp

Xây dựng ứng dụng Java Application thực hiện các chức năng : Lưu trữ thông tin của doctor , tìm thông tin của doctor dựa vào DoctorCode và in danh sách tất cả các doctor ra màn hình, có sử dụng Generic trong chương trình .

```

!!!Welcome to LifeLine Hospital!!!

Select the operation you want to perform:
*****
1. Register Doctor
2. Display the Doctor details
3. Publish the entire list
4. Exit
*****

1
Enter the Doctor code:
123
Enter the name of the doctor:
tomy
Enter the specialization of the Doctor:
abc
Enter the availability of the Doctor in hours:
12

```

Bước 1 . Mở ứng dụng NetBean , New Project | Java | Java Application , nhấn Next , trên hộp thoại “New Java Application ” điền các thông tin trong hộp thoại như sau :
Project Name : **ASM_Ngay_1**; project Location : “chọn đường dẫn lưu ứng dụng” , nhấn Finish để thúc .

- 1.1 Nhấp phải chuột trên **<default package>** chọn New | Java Class , đặt tên lớp : **DoctorDetails**, nhấn Finish để kế thúc .
- 1.2 Nhấp phải chuột trên **<default package>** chọn New | Java Class , đặt tên lớp : **DoctorHash**, nhấn Finish để kế thúc.
- 1.3 Nhấp phải chuột trên **<default package>** chọn New | Java Class, đặt tên lớp : **DoctorHashTest**, nhấn Finish để kế thúc.

Bước 2. Phát triển các lớp để thực hiện các chức năng .

2.1 Phát triển lớp **DoctorDetails**, lớp này mô tả thông tin chi tiết cho doctor. Viết mã cho tập tin **DoctorDetails.java** như sau :

```
public class DoctorDetails {
    //Khai bao cac thanh phan du lieu cho Doctor
    String name;
    String specialization;
    int hours;
    String docCode;
    //Khai bao constructor mac dinh
    public DoctorDetails() {
    }
    //Khai bao constructor co tham so
    public DoctorDetails(final String code, final String lastname,
        final String speciality, final int hours) {
        this.docCode = code;
        this.name = lastname;
        this.specialization = speciality;
        this.hours = hours;
    }
    //Khai bao phuong thuc tra ve thong tin cua doctor
    public String toString() {
        return "\nDoctor code = " + docCode + "\nName = " + name +
            "\nSpecialization = " + specialization + "\nAvailability = " + hours;
    }
}
```

2.2 Phát triển lớp **DoctorHash** , lớp này thực hiện các thao tác : lưu trữ danh sách các doctor , thêm , tìm kiếm thông tin doctor và in danh sách các doctor ra màn hình . Viết mã cho tập tin **DoctorHash.java** như sau :

```
import java.util.*;

public class DoctorHash {
    //Khai bao tap hop luu danh sach cac Doctor
    HashMap<String, DoctorDetails> DoctorHashMap;
    public DoctorHash() {
        //Khoi tao tap hop HashMap
        DoctorHashMap = new HashMap<String, DoctorDetails>();
    }
}
```

```
//Khai bao phuong thuc them 1 doctor vao HashMap
void add() {
    String doctorName = "";
    String specialization = "";
    int hours = 0;
    String code = "";
    // Nhap du lieu tu ban phim
    Scanner input = new Scanner(System.in);
    System.out.println("Enter the Doctor code:");
    code = input.nextLine();

    System.out.println("Enter the name of the doctor:");
    doctorName = input.nextLine();

    System.out.println("Enter the specialization of the Doctor:");
    specialization = input.nextLine();

    System.out.println("Enter the availability of the Doctor in hours:");
    hours = input.nextInt();
    //Tao doi tuong DoctorDetail
    DoctorDetails doctorFields = new DoctorDetails(code, doctorName, specialization, hours);
    //Them doi tuong DoctorDetail vao trong tap hop DoctorHashMap
    DoctorHashMap.put(code, doctorFields);
}

//Khai bao ham hien thi thong tin chi tiet cho 1 doctor
void display() {
    String searchCode = "";
    //Nhap Doctor Code tu ban phim
    Scanner input = new Scanner(System.in);
    System.out.println("Enter the Doctor code:");
    searchCode = input.nextLine();
    //Tim thong tin Doctor theo DoctorCode da nhap
    if(DoctorHashMap.containsKey(searchCode)) {
        //lay thong tin DoctorDetail
        DoctorDetails docObj = DoctorHashMap.get(searchCode);
        System.out.println(docObj);
    }
    else{
        System.out.println("Doctor not found!!");
    }
}
```

```
//Khai bao phuong thuc xuat danh sach cac doctor trong tap hop
void publish() {
    // Display the header information
    System.out.println( "The size of DoctorHashMap = " + DoctorHashMap.size());
    System.out.println( "The Doctors in the Panel are:" );

    Set set = DoctorHashMap.keySet();
    Iterator iter = set.iterator();
    int i = 1;
    while (iter.hasNext()) {
        System.out.println(" " + i + " ) " + DoctorHashMap.get(iter.next()));
        i++;
    }
}
}
```

2.3 Phát triển lớp **DoctorHashTest**, lớp này hiển thị một menu cho người dùng chọn các chức năng để thao tác : thêm , tìm kiếm thông tin doctor và in danh sách các doctor . Viết mã cho tập tin **DoctorHashTest.java** như sau :

```
import java.util.Scanner;

public class DoctorHashTest {
    public static void main(String[] args) {
        //Khai bao doi tuong DoctorHash
        DoctorHash doctorHashObj = new DoctorHash();
        int choice;
        Scanner input = new Scanner(System.in);
        do {
            //Xuat menu de user chon lua cac chuc nang
            System.out.println("\n!!!Welcome to LifeLine Hospital!!!\n" +
                "\nSelect the operation you want to perform:");
            System.out.println("*****");
            System.out.println("1. Register Doctor");
            System.out.println("2. Display the Doctor details");
            System.out.println("3. Publish the entire list");
            System.out.println("4. Exit");
            System.out.println("*****");
        } while (true);
    }
}
```

```

//Nhập ma so tu 1-4 tuong ung 4| chuc nang trong menu
choice = input.nextInt();
switch(choice) {
    case 1: doctorHashObj.add();
        break;
    case 2: doctorHashObj.display();
        break;
    case 3: doctorHashObj.publish();
        break;
    case 4: break;
    default: System.out.println("Not a valid choice!!!");
}
} while(choice != 4);
- }
}

```

Bước 3: Chạy ứng dụng và kiểm tra kết quả

- Nhập vào số 1 , để thêm thông tin doctor .
- Nhập vào số 3 , để xem danh sách các doctor .
- Nhập vào số 2 , để tìm thông tin doctor theo DoctorCode
- Nhập vào số 4 , để thoát ứng dụng.

```

!!!Welcome to LifeLine Hospital!!!

Select the operation you want to perform:
*****
1. Register Doctor
2. Display the Doctor details
3. Publish the entire list
4. Exit
*****

1
Enter the Doctor code:
123
Enter the name of the doctor:
tomy
Enter the specialization of the Doctor:
abc
Enter the availability of the Doctor in hours:
12

```

```

!!!Welcome to LifeLine Hospital!!!

Select the operation you want to perform:
*****
1. Register Doctor
2. Display the Doctor details
3. Publish the entire list
4. Exit|
*****

3
The size of DoctorHashMap = 1
The Doctors in the Panel are:
1 )
Doctor code = 123
Name = tomy
Specialization = abc
Availability = 12

```