

LAB 7: ENUMERATIONS, AUTOBOXING, STATIC IMPORT, AND ANNOTATIONS

MỤC TIÊU:

Kết thúc bài thực hành này bạn có khả năng

- ✓ Sử dụng Enum
- ✓ Sử dụng Boxing/Unboxing
- ✓ Sử dụng Static import
- ✓ Sử dụng Annotation

PHẦN I

BÀI 1 (2 ĐIỂM)

1. Khai báo enum Career mô tả kiểu ngành học tại Fpoly

```
public enum Career {  
    UDPM, TKTW, LTĐĐ, TKĐH  
}
```

2. Khai báo lớp PolyStudent sử dụng Career

```
public class PolyStudent {  
    public String fullname;  
    public Career career;  
  
    public void print() {  
        System.out.println(">Fullname: " + this.fullname);  
        switch (this.career) {  
            case UDPM:  
                System.out.println(">Career: Ứng dụng phần mềm");  
                break;  
            case TKTW:  
                System.out.println(">Career: Thiết kế trang web");  
                break;  
        }  
    }  
}
```

```

        case LTĐĐ:
            System.out.println(">Career: Lập trình di động");
            break;
        case TKĐH:
            System.out.println(">Career: Thiết kế đồ họa");
            break;
    }
}

```

3. Viết phương thức main() sử dụng PolyStudent

```

public static void main(String[] args) {
    PolyStudent sv = new PolyStudent();
    sv.fullname = "Nguyễn Văn Tèo";
    //sv.career = Career.UDPM;
    sv.career = Career.valueOf("UDPM");

    sv.print();
}

```

BÀI 2 (2 ĐIỂM)

Tạo file **Lab7Bai2JFrame.java** và thực hiện theo yêu cầu:

- ✓ Nhập vào 2 chuỗi ký tự
- ✓ Chuyển đổi 2 chuỗi đó thành số
- ✓ Sử dụng try... catch để xử lý ngoại lệ khi không thể chuyển đổi thành số
- ✓ Tính tổng 2 số và in kết quả ra màn hình.

HƯỚNG DẪN:

- ✓ Thiết kế form gồm 3 ô nhập và một nút
- ✓ Viết mã cho nút
 - Đọc dữ liệu từ 2 ô nhập đầu
 - Chuyển đổi sang số thực với Double.parseDouble()

- Tính tổng
- Hiển thị tổng lên ô nhập thứ 3
- Sử dụng JOptionPane.showMessageDialog() để thông báo lỗi

PHẦN II

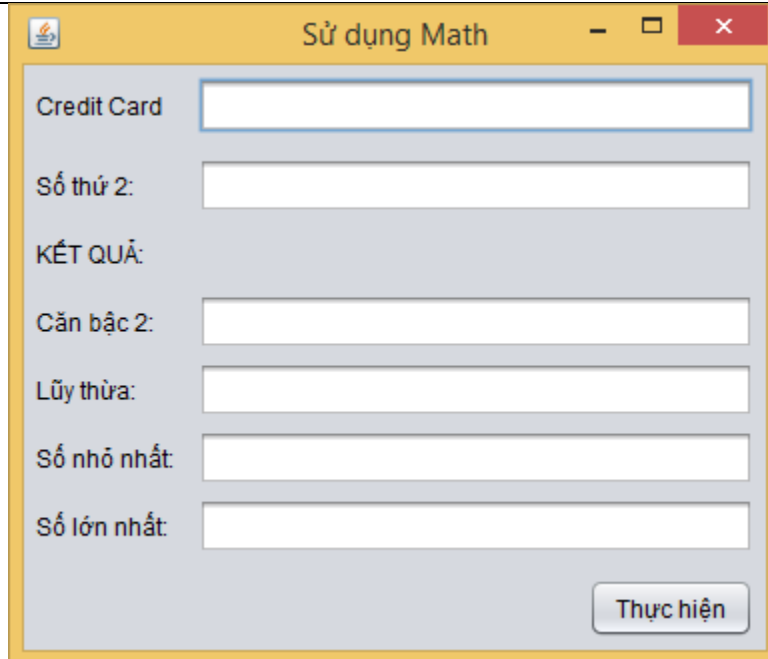
BÀI 3 (2 ĐIỂM)

Tạo file **Lab7Bai3JFrame.java** sử dụng các phương thức trong class Math, thực hiện các công việc sau:

- ✓ Nhập vào hai số nguyên a và b
- ✓ Tính căn bậc 2 của số a
- ✓ Tính a mũ b
- ✓ Tính max giữa 2 số a và b
- ✓ Tính min giữa 2 số a và b

Hướng dẫn

- ✓ Thiết kế form như sau để thực hiện



- ✓ Chú ý:
 - Các ô nhập phần KẾT QUẢ là readonly
 - Kiểm lỗi nhập dữ liệu

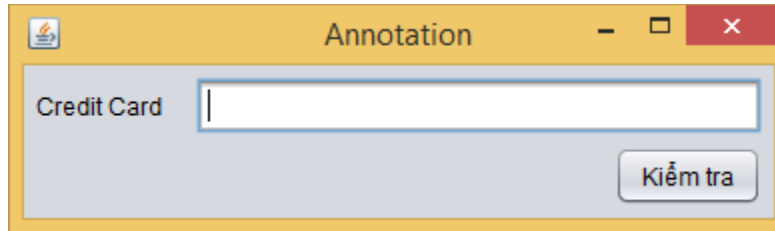
BÀI 4 (2 ĐIỂM)

Tạo file **XPoly.java** và thực hiện theo yêu cầu:

- ✓ Viết phương thức **public static boolean isCardNumber(String number)** thực hiện các công việc:
 - Tính tổng các số có trong dãy số thẻ sinh viên đó
 - Ví dụ: xâu ký tự là: 0090 8767 3224, tổng
 $0+0+9+0+8+7+6+7+3+2+2+4 = 48$
 - Nếu tổng là số chẵn thì trả về true, nếu là số lẻ thì trả về false.
- ✓ Sử dụng annotation **Deprecated** cho phương thức này
- ✓ Tạo form gồm một ô nhập Credit Card và một nút. Tiến hành kiểm tra chuỗi nhập vào có phải là số hợp lệ hay không và đưa ra thông báo thích hợp

Hướng dẫn:

- ✓ Giao diện



- ✓ Tính tổng các số trong chuỗi chứa các ký tự số

```
String s = ...;
int n= Integer.parseInt(s);
int tong = 0;
while(n > 0){
    int a = n % 10;
    n = n/10;
    tong += a;
}
```

BÀI 5 (2 ĐIỂM)

Giảng viên cho thêm