

## CHƯƠNG 7 - XỬ LÝ BIỆT LỆ (EXCEPTION HANDLING)

### I. Tóm tắt lý thuyết

- Biệt lệ (exception) là lỗi xảy ra trong thời gian chương trình thực thi.
- Biệt lệ phải được bắt và xử lý, nếu không chương trình sẽ bị kết thúc đột ngột.
- Dùng khối try...catch để bắt biệt lệ và xử lý lỗi.
- Một khối try có một hoặc nhiều khối catch. Mỗi khối catch chịu trách nhiệm bắt một đối tượng biệt lệ tương ứng được "ném" ra từ khối try. Biệt lệ làm đối số của khối catch đặt sau không được là lớp con của biệt lệ làm đối số của khối catch đã đặt trước.
- Một khối try có duy nhất một khối finally được đặt sau tất cả các khối catch và luôn luôn được thực hiện có hay không có lỗi xảy ra.
- Người dùng có thể tự tạo một lớp biệt lệ bằng cách dẫn xuất từ lớp Exception.
- Có thể xử lý biệt lệ ngay trong phương thức dùng khối try...catch hoặc chuyển việc xử lý biệt lệ ra ngoài bằng cách dùng từ các từ khóa throw để ném đối tượng biệt lệ ra ngoài.
- Một hàm có khai báo ném biệt lệ ra ngoài (dùng từ khóa throws) phải được gọi trong khối try...catch.

### II. Bài tập thực hành mẫu

1. Viết chương trình cho phép người dùng nhập một chuỗi số nguyên từ bàn phím, sau đó chuyển chuỗi nhận được sang số nguyên. Trong trường hợp chuỗi nhập vào không đúng định dạng, chương trình sẽ không thể chuyển chuỗi sang dạng số thì hiển thị một thông điệp thông báo lỗi.

```
import java.util.*;
public class ParseInteger
{
    public static void main(String args[])
    {
        try
        {

            /*Nhập chuỗi số nguyên từ bàn phím*/
            Scanner kb = new Scanner(System.in);
            System.out.print("Nhập chuỗi số nguyên: ");
            String str=kb.nextLine();

            /*Chuyển chuỗi sang số nguyên, nếu chuỗi không đúng định dạng, hàm
```

```

parseInt sẽ phát ra một đối tượng biệt lệ*/

    int num = Integer.parseInt(str);

    /*Hiển thị kết quả*/
    System.out.println(num);
}
catch(NumberFormatException e)
{

    /*Nếu chuỗi nhập vào không đúng định dạng, hàm parseInt không chuyển
    được sang số nguyên thì một đối tượng lỗi NumberFormatException được
    phát ra và khối catch bắt lấy và được thực hiện*/

    System.out.println("Chuỗi nhập vào không đúng định dạng");
}
}
}

```

Kết quả chương trình:

```

<terminated> ParseInteger [Java Application] C:\Program Files\Java\jdk1.7.0_45\jre\bin\javaw.exe (Nov 18, 2013, 9:35:29 PM)
Nhập chuỗi số nguyên: A123
Chuỗi nhập vào không đúng định dạng

```

2. Xây dựng lớp EmailException dùng để biểu diễn lỗi địa chỉ email không hợp lệ. Viết chương trình nhập địa chỉ email từ bàn phím, kiểm tra địa chỉ email nếu không hợp lệ thì hiển thị thông báo lỗi.

Đáp án 1:

```

import java.util.Scanner;

/*Tạo Lớp EmailAddressException kế thừa Lớp Exception*/
class EmailAddressException extends Exception
{

    /*Định nghĩa hàm khởi tạo*/
    EmailAddressException(String str)
    {

        /*gọi hàm khởi tạo của Lớp Exception*/
        super(str);
    }
}

/*Tạo Lớp kiểm tra địa chỉ email*/
public class EmailCheck
{

    /* Hàm kiểm tra địa chỉ email. Giả sử quy tắc kiểm tra đơn giản như sau:

```

1. Địa chỉ email chỉ chứa duy nhất một ký tự @.
2. Ký tự @ không thể ở vị trí đầu tiên. \*/

```

public static void check(String str) throws EmailAddressException
{
    /*Kiểm tra chuỗi có chứa @*/
    int place = str.indexOf('@');

    /*Nếu không chứa @ */
    if(place==-1)
    {

        /*Ném ra một đối tượng EmailAddressException*/
        throw new EmailAddressException("Khong hop le do khong chua '@'");
    }

    /*Nếu kí tự đầu tiên của chuỗi là @ */
    if(place==0)
    {
        throw new EmailAddressException("Khong hop le do '@' o vi tri dau tien ");
    }

    /*Nếu có chứa @, tiếp tục kiểm tra có chứa @ khác không*/
    place = str.indexOf('@',place+1);

    /*Nếu tìm thấy thêm @ trong chuỗi*/
    if(place!=-1)
    {
        throw new EmailAddressException("Khong hop le do co hon 1 '@'");
    }

    /*Không rơi vào các trường hợp trên thì địa chỉ hợp lệ*/
    System.out.println("Dia chi email hop le");
}

public static void main(String args[])
{
    try
    {

        /*Nhập chuỗi địa chỉ email từ bàn phím*/
        Scanner kb = new Scanner(System.in);
        System.out.print("Nhap dia chi email: ");
        String email=kb.nextLine();

        /*gọi hàm check để kiểm tra, vì hàm throws ra đối tượng Exception nên phải gọi trong khối try...catch*/

        check(email);
    }
    catch(EmailAddressException e)
    {

        /*Nếu chuỗi địa chỉ email không hợp lệ thì hàm check ném ra đối tượng EmailAddressException và khối catch bắt lấy và được thực thi*/

        System.out.println(e.getMessage());
    }
}

```

```
}
}}
```

Kết quả chương trình:

```
<terminated> EmailCheck (1) [Java Application] C:\Program Files\Java\jdk1.7.0_45\jre\bin\javaw.exe (Nov 18, 2013, 9:45:44 PM)
Nhap dia chi email: hcp
khong hop le do khong chua '@'
```

Đáp án 2: Sử dụng biểu thức chính qui (Regular expression) để viết lại hàm check() kiểm tra địa chỉ email một cách đầy đủ và chính xác hơn.

```
/*Định nghĩa lại hàm check sử dụng biểu thức chính qui*/
public static void check(String str) throws EmailAddressException
{
    /*Biểu thức chính quy so khớp đ/c email */
    String emailreg = "^([\\w- ]+@([\\w- ]+\\.)+[\\w- ]+)$";

    /*So khớp địa chỉ email với biểu thức chính qui*/
    Boolean b = str.matches(emailreg);

    /*Nếu không khớp biểu thức chính qui thì sinh ra lỗi*/
    if (b == false) throw new EmailAddressException("Dia chi email khong
hop le ");
}
```

3. Xây dựng lớp UnsignedIntegerException dùng để biểu diễn lỗi định dạng số nguyên dương không hợp lệ. Viết chương trình nhập một chuỗi từ bàn phím, kiểm tra nếu chuỗi không đúng định dạng số nguyên dương thì hiển thị thông báo lỗi.

```
import java.util.Scanner;

/*Xây dựng Lớp UnsignedIntegerException kế thừa Lớp Exception*/
class UnsignedIntegerException extends Exception
{
    /*Hàm khởi tạo*/
    public UnsignedIntegerException(String str)
    {
        /*Gọi hàm khởi tạo Lớp cha Exception*/
        super(str);
    }
}
```

```

    }
}

/*Tạo Lớp kiểm tra định dạng số*/
public class CheckNumber
{
    /*Định nghĩa hàm checkNumber để kiểm tra định dạng số*/
    public static void checkNumber(String s) throws UnsignedIntegerException
    {
        /*Tạo biểu thức chính quy để định nghĩa định dạng chuỗi số nguyên*/
        String numReg = "^\\d+$";

        /*So khớp chuỗi với biểu thức chính quy*/
        Boolean b = s.matches(numReg);

        /*Nếu không khớp thì ném ra đối tượng UnsignedIntegerException*/
        if (b == false) throw new UnsignedIntegerException("Dinh dang
chuoi so nguyen duong khong hop le");
    }

    public static void main(String args[])
    {
        try
        {
            /*Nhập chuỗi từ bàn phím*/
            Scanner kb = new Scanner(System.in);
            System.out.print("Nhap chuoi so nguyen duong: ");
            String numStr=kb.nextLine();

            /*Gọi hàm checkNumber, nếu chuỗi nhập vào không đúng định dạng thì hàm
            checkNumber sẽ sinh ra một đối tượng lỗi, do đó nó được gọi trong khối
            try...catch*/

            checkNumber(numStr);
        }
        catch(UnsignedIntegerException e)
        {
            /*Nếu có đối tượng Lỗi ném ra từ hàm checkNumber thì khối catch được
            thực thi*/

            System.out.println(e.getMessage());
        }
    }
}

```

Kết quả chương trình:

```
<terminated> CheckNumber [Java Application] C:\Program Files\Java\jdk1.
Nhap chuoi so nguyen duong: 221a2s
Dinh dang chuoi so nguyen duong khong hop le
```

4. Xây dựng lớp `WebAddressException` dùng để biểu diễn lỗi định dạng địa chỉ trang web không hợp lệ. Viết chương trình nhập địa chỉ trang web, kiểm tra nếu không đúng định dạng thì thông báo lỗi.

```
import java.util.Scanner;

/*Tạo Lớp WebAddressException từ Lớp Exception*/
class WebAddressException extends Exception
{

/*Hàm khởi tạo*/
public WebAddressException(String str)
{

/*Gọi hàm khởi tạo của Lớp cha*/
super(str);
}

}

/*Tạo Lớp để kiểm tra địa chỉ web*/
public class WebAddressCheck
{

/*Hàm kiểm tra địa chỉ web*/
public static void checkWebaddress(String s) throws
WebAddressException
{

/*tạo biểu thức chính quy để so khớp*/
String numReg = "^https?:\\/([\\w-]+\\.)+[\\w-]+(\\/([\\w-\\.\\/?%]=)*)?$";

/*So khớp chuỗi với biểu thức chính quy*/
Boolean b = s.matches(numReg);

/*Nếu không khớp sinh ra đối tượng Lỗi*/
if (b == false) throw new WebAddressException("Dia chi web khong
hop le");
else System.out.println("Dia chi web hop le");
}

public static void main(String args[])
{
try
{

/*Nhập địa chỉ web từ bàn phím*/
Scanner kb = new Scanner(System.in);
System.out.print("Nhap dia chi web: ");
```

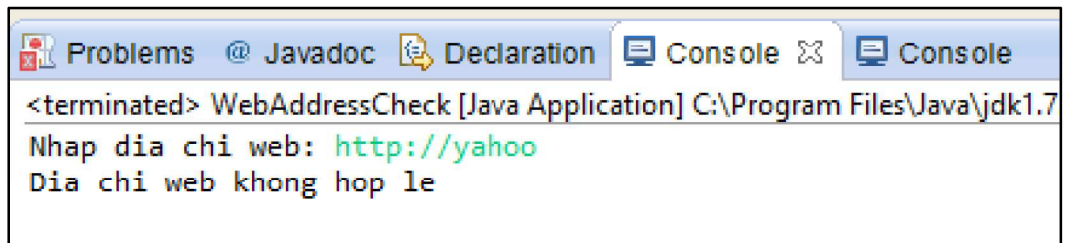
```
String webadd =kb.nextLine();

/*Gọi hàm để kiểm tra địa chỉ web nhập vào, nếu không đúng định dạng
thì sinh ra đối tượng Lỗi*/

    checkWebaddress(webadd);
}
catch(WebAddressException e)
{

    /*Nếu hàm checkWebaddress sinh ra Lỗi thì khối catch thực thi*/
    System.out.println(e.getMessage());
}
}
}
```

Kết quả chương trình:



5. Xây dựng lớp FractionException dùng để biểu diễn lỗi địa chỉ phân số không hợp lệ. Viết chương trình tạo lớp Fraction biểu diễn một phân số và thực hiện phép chia hai phân số. Trường hợp mẫu số có giá trị 0, chương trình sẽ sinh ra một đối tượng biệt lệ FractionException

```
/*Tạo Lớp FractionException kế thừa Lớp Exception*/
class FractionException extends Exception
{

    /*Hàm khởi tạo*/
    public FractionException (String str)
    {

        /*Gọi hàm khởi tạo của Lớp cha*/
        super(str);
    }
}

/* Tạo Lớp để kiểm tra phân số*/
public class Fraction
{

    /*Khai báo 2 thuộc tính tử và mẫu*/
    private int numerator;
    private int denominator;
```

```

/*Hàm khởi tạo*/
public Fraction(int num1, int num2) throws FractionDivideByZeroException
{

    /*Nếu mẫu số = 0 thì sinh ra Lỗi*/
    if(num2==0)
    {

        /*Ném ra đối tượng Lỗi*/
        throw new FractionDivideByZeroException("Loi do mau so = 0");
    }

    /*Nếu hợp lệ thì khởi tạo các thuộc tính*/
    numerator = num1;
    denominator = num2;
}

/*Phương thức trả lại chuỗi nội dung của phân số*/
public String toString()
{
    return numerator + "/" + denominator;
}

/* Hàm chia hai phân số*/
public static Fraction divide(Fraction frac1, Fraction frac2)
    throws FractionDivideByZeroException
{
    /*Nếu tử của phân số thứ 2 là 0 thì kết quả của chia hai phân số sẽ có
    mẫu là 0*/
    if(frac2.numerator==0)
    {

        /*Sinh ra Lỗi*/
        throw new FractionDivideByZeroException("Loi do chia cho 0");
    }

    /*Nếu không rơi trường hợp trên thì tạo biểu thức kết quả*/
    return new Fraction(frac1.numerator*frac2.denominator,
    frac1.denominator*frac2.numerator);
}

public static void main(String args[])
{

    /*Ba phân số*/
    Fraction fr1, fr2, fr3;
    try
    {

        /*Khởi tạo giá trị cho ba phân số*/
        fr1 = new Fraction(1,2);
        fr2 = new Fraction(0,1);

        /*Gọi hàm chia hai phân số*/
        Fraction temp = divide(fr1,fr2);
        System.out.println(temp);
    }
    catch(FractionDivideByZeroException e)
    {

```

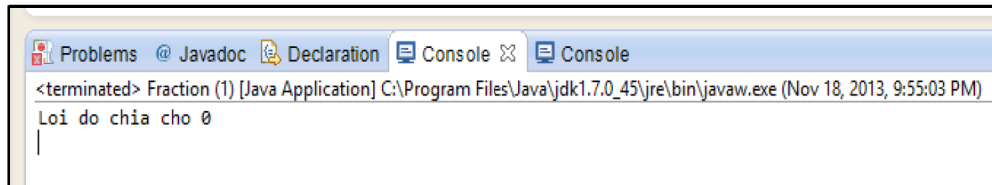


```

        System.out.println(e.getMessage());
    }
}

```

Kết quả chương trình:



### III. Bài tập tự giải

1. Xây dựng lớp `AgeException` biểu diễn lỗi giá trị tuổi không hợp lệ. Viết chương trình nhập vào một tuổi, nếu giá trị tuổi không hợp lệ thì chương trình thông báo lỗi.
2. Xây dựng lớp `PINException` biểu diễn lỗi định dạng số PIN (giả sử định dạng chuỗi số PIN hợp lệ là chuỗi có 4 ký tự số). Viết chương trình nhập chuỗi và thông báo lỗi nếu chuỗi nhập không đúng định dạng số PIN. Gợi ý: sử dụng biểu thức chính quy có dạng: `^\d{4}$`
3. Xây dựng lớp `IntegerException` dùng để biểu diễn lỗi định dạng số nguyên không hợp lệ. Viết chương trình nhập một chuỗi từ bàn phím, kiểm tra nếu chuỗi không đúng định dạng số nguyên thì hiển thị thông báo lỗi. Gợi ý: Sử dụng biểu thức chính quy: `^-?\d+$`
4. Xây dựng lớp `PhoneException` biểu diễn lỗi số điện thoại không hợp lệ. Viết chương trình nhập vào một số điện thoại (dạng chuỗi số), nếu số điện thoại không hợp lệ thì chương trình thông báo lỗi.
5. Xây dựng lớp `TriangleEdgesException` biểu diễn lỗi giá trị cạnh của tam giác không hợp lệ. Viết chương trình nhập 3 cạnh của một tam giác, nếu cạnh không hợp lệ thì chương trình thông báo lỗi.