





B môn Công nghệ Phần mềm
Viện CNTT & TT
Trường Đại học Bách Khoa Hà Nội

LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG


Bài 08. Ngo il và x lý ngo il



M c tiêu c a bài h c

- Gi i thích v ngo il là gì và mô t các l i ích c a vi c x lý ngo il h ng it ng
- Gi i thích c mô hình x lý ngo il
- S d ng kh i try/catch/finally b t và x lý ngo il trong Java
- Hi u và bi t cách s d ng y nhi m ngo il
- Bi t cách t o ra và s d ng ngo il t nh ngh a


2



N i dung

1. Ngo il
2. B t và x lý ngo il
3. y nhi m ngo il
4. T o ngo il t nh ngh a


3



N i dung


1. Ngo il
2. B t và x lý ngo il
3. y nhi m ngo il
4. T o ngo il t nh ngh a

4




1.1. Ngo il là gì?

- Exception = Exceptional event

Ví d : $4/0 =$  ERROR!!

5



1.1. Ngo il là gì? (2)

```
float number1,number2;
//input number1,number2;
float division= number1/number2;
```

No handler exists

EXIT

6

1.2. Cách xử lý lỗi truy cập

- Vì mã xử lý lỗi phát sinh ra lỗi
- Truy cập trạng thái lên mức trên

7

Ví dụ

```
int divide(int num, int denom, int *error)
{
    if (denom != 0){
        error = 0;
        return num/denom;
    } else {
        error = 1;
        return 0;
    }
}
```

8

Nhắc lại

9

Nội dung

1. Ngồi
2. Bắt và xử lý ngoại lệ
3. Ý nghĩa ngoại lệ
4. Ngoại lệ trong thực tế

10

2.1. Mục đích của xử lý ngoại lệ

```
.....
IF B IS ZERO GO TO ERROR
C = A/B
PRINT C
GO TO EXIT
```

```
ERROR:
  DISPLAY "DIVISION BY ZERO"
```

```
EXIT:
  END
```

} Khối xử lý lỗi

11

2.1. Mục đích của xử lý ngoại lệ (2)

- Khi xảy ra ngoại lệ, nếu không có cách xử lý thích hợp?

12

2.2. Mô hình xử lý ngoại lệ

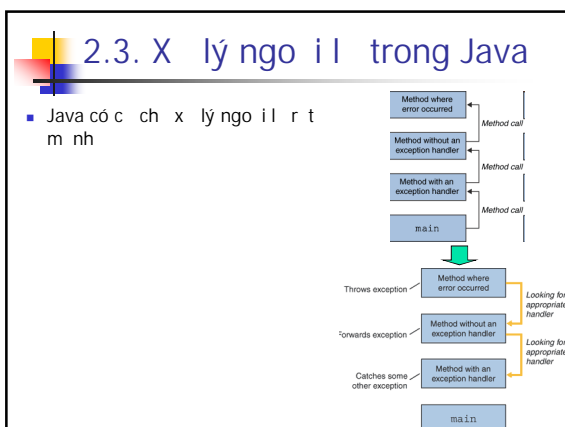
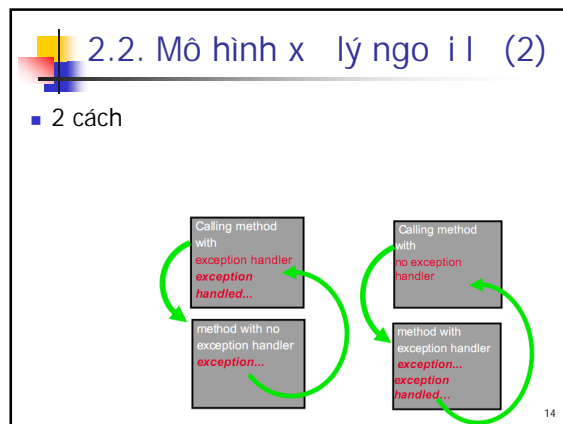
- Hình thức

```

float sales = getSales();
int staffsize = getStaff().size();
float avg_sales = sales/staffsize;
System.out.println(avg_sales);

```

13



2.3. Xử lý ngoại lệ trong Java (2)

- Các từ khóa
 - try
 - catch
 - finally
 - throw
 - throws

16

2.3.1. Khi try/catch

- Khi try ... catch:


```

try {
    // Đoạn mã có thể gây ngoại lệ
}
catch (ExceptionType e) {
    // Xử lý ngoại lệ
}

```

17

Ví dụ không xử lý ngoại lệ

```

class NoException {
    public static void main(String args[]) {
        String text = args[0];
        System.out.println(text);
    }
}

```

18

Ví dụ có xử lý ngoại lệ

```
class ArgExceptionDemo {
    public static void main(String args[]) {
        try {
            String text = args[0];
            System.out.println(text);
        }
        catch(Exception e) {
            System.out.println("Hay nhap tham so khi chay!");
        }
    }
}
```

19

Ví dụ chia cho 0

```
public class ChiaCho0Demo {
    public static void main(String args[]){
        try {
            int num = calculate(9,0);
            System.out.println(num);
        }
        catch(Exception e) {
            System.err.println("Co loi xay ra: " + e.toString());
        }
    }
    static int calculate(int no, int nol){
        int num = no / nol;
        return num;
    }
}
```

20

2.3.2 Cây phân cấp ngoại lệ trong Java

21

a. Lớp Throwable

- Mối quan hệ giữa các lớp?

22

```
public class StackExceptionDemo {
    public static void main(String args[]){
        try {
            int num = calculate(9,0);
            System.out.println(num);
        }
        catch(Exception e) {
            System.err.println("Co loi xay ra :"+
                               e.getMessage());
            e.printStackTrace();
        }
    }
    static int calculate(int no, int nol) {
        int num = no / nol;
        return num;
    }
}
```

23

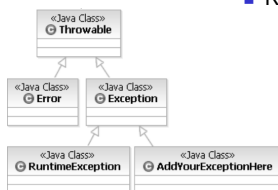
b. Lớp Error

- Các lớp con:
 - VirtualMachineError: InternalError, OutOfMemoryError, StackOverflowError, UnknownError
 - ThreadDeath
 - LinkageError:
 - IncompatibleClassChangeError
 - AbstractMethodError, InstantiationException, NoSuchFieldError, NoSuchMethodError...
 - ...
 - ...

24

c. L p Exception

- Ch a các lo i ngo i l
nên/ph i b t và x lý
h o c y nhi m.
- RuntimeException?



25

M t s l p con c a Exception

- ClassNotFoundException, SQLException
- java.io.IOException:
 - FileNotFoundException, EOFException...
- RuntimeException:
 - NullPointerException, BufferOverflowException
 - ClassCastException, ArithmeticException
 - IndexOutOfBoundsException:
 - ArrayIndexOutOfBoundsException,
 - StringIndexOutOfBoundsException...
 - IllegalArgumentException:
 - NumberFormatException, InvalidParameterException...
 - ...

26

Ví d IOException

```

import java.io.InputStreamReader;
import java.io.IOException;
public class HelloWorld{
    public static void main(String[] args) {
        InputStreamReader isr = new
            InputStreamReader(System.in);
        try {
            System.out.print("Nhập vào 1 ký tự: ");
            char c = (char) isr.read();
            System.out.println("Ký tự vừa nhập: " + c);
        } catch (IOException ioe) {
            ioe.printStackTrace();
        }
    }
}
  
```

27

2.3.3. Kh i try – catch l ng nhau

```

try {
    // Doan ma co the gay ra IOException
    try {
        // Doan ma co the gay ra NumberFormatException
        catch (NumberFormatException e1) {
            // Xu ly loi sai dinh dang so
        }
    } catch (IOException e2) {
        // Xu ly loi vao ra
    }
}
  
```

28

2.3.4. Nhi u kh i catch

```

try {
    // Doan ma co the gay ra nhieu ngoai
    le
} catch (ExceptionType1 e1) {
    // Xu ly ngoai le 1
} catch (ExceptionType2 e2) {
    // Xu ly ngoai le 2
} ...
  
```

29

```

class MultipleCatch1 {
    public static void main(String args[])
    {
        try {
            String num = args[0];
            int numValue = Integer.parseInt(num);
            System.out.println("Diện tích hv là: "
                + numValue * numValue);
        } catch (Exception e1) {
            System.out.println("Hay nhập cạnh của hv!");
        } catch (NumberFormatException e2) {
            System.out.println("Not a number!");
        }
    }
}
  
```

30

```

class MultipleCatch1 {
    public static void main(String args[])
    {
        try {
            String num = args[0];
            int numValue = Integer.parseInt(num);
            System.out.println("Dien tich hv la: "
                               + numValue * numValue);
        } catch(ArrayIndexOutOfBoundsException e1) {
            System.out.println("Hay nhap canh cua hv!");
        } catch(NumberFormatException e2){
            System.out.println("Hay nhap l so!");
        }
    }
}

```

31

```

class MultiCatch2 {
    public static void main( String args[]) {
        try {
            // format a number
            // read a file
            // something else...
        }
        catch(IOException e) {
            System.out.println("I/O error "+e.getMessage());
        }
        catch(NumberFormatException e) {
            System.out.println("Bad data "+e.getMessage());
        }
        catch(Throwable e) { // catch all
            System.out.println("error: " + e.getMessage());
        }
    }
}

```

32

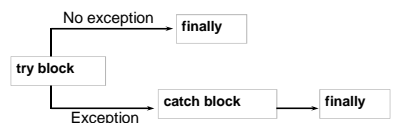
```

...
public void openFile(){
    try {
        // constructor may throw FileNotFoundException
        FileReader reader = new FileReader("someFile");
        int i=0;
        while(i != -1) {
            //reader.read() may throw IOException
            i = reader.read();
            System.out.println((char) i );
        }
        reader.close();
        System.out.println("--- File End ---");
    } catch (FileNotFoundException e) {
        //do something clever with the exception
    } catch (IOException e) {
        //do something clever with the exception
    }
}
...

```

33

2.3.5. Khi i finally



34

Cú pháp try ... catch ... finally

```

try {
    // Khoi lenh co the sinh ngoai le
}
catch(ExceptionType e) {
    // Bat va xu ly ngoai le
}
finally {
    /* Thuc hien cac cong viec can thiet du
    ngoai le co xay ra hay khong */
}

```

35

```

class StrExceptionDemo {
    static String str;
    public static void main(String s[]) {
        try {
            System.out.println("Truoc ngoai le");
            staticLengthMethod();
            System.out.println("Sau ngoai le");
        }
        catch(NullPointerException ne) {
            System.out.println("Da xay ra loi");
        }
        finally {
            System.out.println("Trong finally");
        }
    }

    static void staticLengthMethod() {
        System.out.println(str.length());
    }
}

```

36

```

public void openFile(){
    try {
        // constructor may throw FileNotFoundException
        FileReader reader = new FileReader("someFile");
        int i=0;
        while(i != -1) {
            //reader.read() may throw IOException
            i = reader.read();
            System.out.println((char) i );
        } catch (FileNotFoundException e) {
            //do something clever with the exception
        } catch (IOException e) {
            //do something clever with the exception
        } finally {
            reader.close();
            System.out.println("--- File End ---");
        }
    }
}

```

37

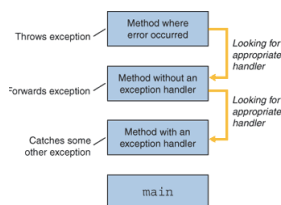
Nội dung

1. Ngồi
2. Bớt và xử lý ngồi
3. ý nghĩa ngồi
4. Tới ngồi tới nhàn hạ

38

Hai cách làm việc với ngoại lệ

- Xử lý ngay
- Ý nghĩa cho vị trí gọi nó:



39

3.1. Ý nghĩa ngồi

- Ví dụ

```

public void myMethod(int param) throws
Exception{
    if (param < 10) {
        throw new Exception("Too low!");
    }
    //Blah, Blah, Blah...
}

```

40

3.1. Ý nghĩa ngồi (3)

- Ví dụ

```

class Test {
    public void myMethod(int param) {
        if (param < 10) {
            throw new RuntimeException("Too low!");
        }
        //Blah, Blah, Blah...
    }
}

```

- → Không hiểu?

41

```

public class DelegateExceptionDemo {
    public static void main(String args[]){
        int num = calculate(9,3);
        System.out.println("Lan 1: " + num);
        num = calculate(9,0);
        System.out.println("Lan 2: " + num);
    }
    static int calculate(int no, int nol)
        throws ArithmeticException {
        if (nol == 0)
            throw new
                ArithmeticException("Không thể chia cho 0!");
        int num = no / nol;
        return num;
    }
}

```

42

```

public class DelegateExceptionDemo {
    public static void main(String args[]){
        int num = calculate(9,3);
        System.out.println("Lan 1: " + num);
        num = calculate(9,0);
        System.out.println("Lan 2: " + num);
    }
    static int calculate(int no, int nol)
        throws Exception {
        if (nol == 0)
            throw new
                ArithmeticException("Khong the chia cho 0!");
        int num = no / nol;
        return num;
    }
}

```

43

```

public class DelegateExceptionDemo {
    public static void main(String args[]){
        try {
            int num = calculate(9,3);
            System.out.println("Lan 1: " + num);
            num = calculate(9,0);
            System.out.println("Lan 2: " + num);
        } catch(Exception e) {
            System.out.println(e.getMessage());
        }
    }
    static int calculate(int no, int nol)
        throws ArithmeticException {
        if (nol == 0)
            throw new
                ArithmeticException("Khong the chia cho 0!");
        int num = no / nol;
        return num;
    }
}

```

44

3.1. y nghi m ngo i l (4)

■ y nghi m nghi u h n 1 ngo i l

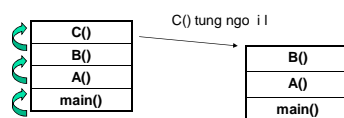
```

public void myMethod(int tuoi, String ten)
    throws ArithmeticException, NullPointerException{
    if (tuoi < 18) {
        throw new ArithmeticException("Chua du tuoi!");
    }
    if (ten == null) {
        throw new NullPointerException("Thieu ten!");
    }
    //Blah, Blah, Blah...
}

```

45

3.2. Lan truy n ngo i l



- N u C() g p l i và tung ra ngo i l nh ng trong C() l i không x lý ngo i l này, thì ch còn m t n i có th x lý chính là n i mà C() c g i, ó là trong ph ng th c B().

■ ...

46

3.3. K th a và y nghi m ngo i l

- Khi override m t ph ng th c c a l p cha, ph ng th c l p con không c phép tung ra các ngo i l m i

47

3.3. K th a và y nghi m ngo i l (2)

```

class Disk {
    void readFile() throws EOFException {}
}
class FloppyDisk extends Disk {
    void readFile() throws IOException {}
}

```



```

class Disk {
    void readFile() throws IOException {}
}
class FloppyDisk extends Disk {
    void readFile() throws EOFException {}
}

```

48

3.4. u i m c a y n h i m n g o i l

49

N i d u n g

1. Ngo i l
2. B t và x lý ngo i l
3. y n h i m n g o i l
4. T o ngo i l t nh ngh a

50

4. T o ngo i l t nh ngh a

```
public class MyException extends Exception {
    public MyException(String msg) {
        super(msg);
    }
    public MyException(String msg, Throwable cause){
        super(msg, cause);
    }
}
```

51

S d ng ngo i l ng i dùng nh ngh a

```
public class FileExample
{
    public void copyFile(String fName1,String fName2)
    throws MyException
    {
        if (fName1.equals(fName2))
            throw new MyException("File trung ten");
        // Copy file
        System.out.println("Copy completed");
    }
}
```

52

S d ng ngo i l ng i dùng nh ngh a

■ B t và x lý ngo i l

```
public class Test {
    public static void main(String[] args) {
        FileExample obj = new FileExample();
        try {
            String a = args[0];
            String b = args[1];
            obj.copyFile(a,b);
        } catch (MyException e1) {
            System.out.println(e1.getMessage());
        }
        catch(Exception e2) {
            System.out.println(e2.toString());
        }
    }
}
```

53