

Exception Handling

"Understanding, Using, and Handling runtime errors "
Advanced in Programming

Shakirullah Waseeb
shakir.waseeb@gmail.com

Nangarhar University

March 7, 2017



Agenda

- 1 Understanding Exception
- 2 Exceptions in Java
 - Introduction
 - Different Scenarios
 - Catching Exceptions; Syntax, and Example
- 3 Exceptions in PHP
 - Exception without catch
 - Exception with try and catch
- 4 Exceptions in Python
 - Exception without try catch
 - Exception with try catch
- 5 Questions and Discussion



What are exceptions?

- **Exception:** errors that rise during programs execution, indication of problem that occurs during a program's execution, or an event that disrupts the normal flow of the program (e.g. divide by zero)
- **Exception Handling:** mechanism to handle such problems and errors e.g. `IndexOutOfBoundsException`, `IO`, `SQL`, `Remote` etc.
- Core advantage is to *maintain the normal flow of the application*
- There are mainly two types of exceptions:
 - Checked exceptions: exceptions that occurs at compile time, can not be ignored (e.g. `IOException`, `SQLException` etc.)
 - Unchecked exceptions: occurs at program execution time also called runtime exceptions, and are ignored at compile time. These include programming bugs, such as logic errors or improper use of an API (e.g. `ArithmeticExceptions`, `NullPointerException` etc.)
 - Errors: These are not exceptions at all, but problems that arise beyond the control of the user or the programmer (e.g. `OutOfMemoryError`, `VirtualMachineError` etc.)



Exception Handling Terminology

- Common terms used by most of the programming languages in exception handling *throw, rise, try, except, catch, finally*
 - **throw**; used to trigger an exception, must have at least one catch
 - **rise**; same as throw statement
 - **try**; a block where an exception may occur and for which a particular exception will be activated
 - **catch**; specify the type of exception to be handled
 - **except**; same as catch statement
 - **finally**; block that run in either condition, whether an exception occur or not



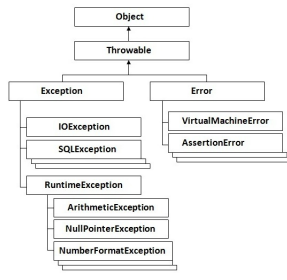
Agenda

- 1 Understanding Exception
- 2 **Exceptions in Java**
 - Introduction
 - Different Scenarios
 - Catching Exceptions; Syntax, and Example
- 3 Exceptions in PHP
 - Exception without catch
 - Exception with try and catch
- 4 Exceptions in Python
 - Exception without try catch
 - Exception with try catch
- 5 Questions and Discussion



Java and Exception handling

- Java Exception Handling Keywords: *try, catch, finally, throw, and throws*
- All exception classes are subtypes of `java.lang.Exception` class
- Exception class is a subclass of the `Throwable` class
- There is another subclass called `Error` which is derived from the `Throwable` class



[1]

Figure: Hierarchy of Java Exception classes



Agenda

- 1 Understanding Exception
- 2 **Exceptions in Java**
 - Introduction
 - **Different Scenarios**
 - Catching Exceptions; Syntax, and Example
- 3 Exceptions in PHP
 - Exception without catch
 - Exception with try and catch
- 4 Exceptions in Python
 - Exception without try catch
 - Exception with try catch
- 5 Questions and Discussion



Common Java exceptions scenario

- Common scenario in which exceptions may occur:

1 ArithmeticException

```
int a=50/0;//ArithmeticException
```

[1]

2 NullPointerException

```
String s=null;  
System.out.println(s.length());//NullPointerException
```

[1]

3 NumberFormatException

```
String s="abc";  
int i=Integer.parseInt(s);//NumberFormatException
```

[1]

4 ArrayIndexOutOfBoundsException

```
int a[]=new int[5];  
a[10]=50; //ArrayIndexOutOfBoundsException
```

[1]



Agenda

- 1 Understanding Exception
- 2 Exceptions in Java
 - Introduction
 - Different Scenarios
 - Catching Exceptions; Syntax, and Example
- 3 Exceptions in PHP
 - Exception without catch
 - Exception with try and catch
- 4 Exceptions in Python
 - Exception without try catch
 - Exception with try catch
- 5 Questions and Discussion



Catching Exception

- A method catches exception using a combination of *try* and *catch* blocks.
- *try/catch* block is placed around the code which might rise an exception
- Code within *try/catch* block is referred to as protected code

Syntax [2]

```
try{  
    // protected code  
} catch (ExceptionName excep){  
    // code that execute in case an exception occur  
}
```



Catching Exception Example

Handling IndexOutOfBoundsException Example [2]

```
// File Name : ExcepTest.java
import java.io.*;
public class ExcepTest {
    public static void main(String args[]) {
        try {
            int var[] = new int[2];
            System.out.println("Access element three :" + var[3]);
        } catch (ArrayIndexOutOfBoundsException e) {
            System.out.println("Exception thrown :" + e);
        }
        System.out.println("Out of the block");
    }
}
```



throw and throws keywords

- *throws*: used to postpone the handling of a checked exception
- *throw*: used to invoke an exception explicitly

Example: throws RemoteException, InsufficientFundsException [2]

```
import java.io.*;
public class Account {
    public void deposit(double amount) throws RemoteException {
        // Method implementation
        throw new RemoteException();
    }
    public void withdraw(double amount) throws RemoteException, InsufficientFundsException {
        // Method implementation
        throw new RemoteException();
        if (this.amount < amount){
            throw new InsufficientFundsException();
        }
    }
    // Remainder of class definition
}
```



User defined exception

User defined InsufficientFundsException Example [2]

```
// File Name InsufficientFundsException.java
import java.io.*;
public class InsufficientFundsException extends Exception {
    private double amount;
    public InsufficientFundsException(double amount) {
        this.amount = amount;
    }
    public double getAmount() {
        return amount;
    }
}
```



Agenda

- 1 Understanding Exception
- 2 Exceptions in Java
 - Introduction
 - Different Scenarios
 - Catching Exceptions; Syntax, and Example
- 3 Exceptions in PHP
 - Exception without catch
 - Exception with try and catch
- 4 Exceptions in Python
 - Exception without try catch
 - Exception with try catch
- 5 Questions and Discussion



PHP: Exception without catch

User defined InsufficientFundsException Example [3]

```
<?php
//create function with an exception
function checkNum($number) {
    if($number > 1) {
        throw new Exception("Value must be 1 or below");
    }
    return true;
}
//trigger exception
checkNum(2);
?>
```

Output:

```
Fatal error: Uncaught exception 'Exception'
with message 'Value must be 1 or below' in C:\webfolder\test.php:6
Stack trace: #0 C:\webfolder\test.php(12):
checkNum(2) #1 {main} thrown in C:\webfolder\test.php on line 6
```

[3]



Agenda

- 1 Understanding Exception
- 2 Exceptions in Java
 - Introduction
 - Different Scenarios
 - Catching Exceptions; Syntax, and Example
- 3 Exceptions in PHP
 - Exception without catch
 - Exception with try and catch
- 4 Exceptions in Python
 - Exception without try catch
 - Exception with try catch
- 5 Questions and Discussion



PHP: Exception with try and catch

User defined InsufficientFundsException Example [3]

```
<?php
//create function with an exception
function checkNum($number) {
    if($number > 1) {
        throw new Exception("Value must be 1 or below");
    }
    return true;
}
//trigger exception in a "try" block
try {
    checkNum(2);
    //If the exception is thrown, this text will not be shown
    echo 'If you see this, the number is 1 or below';
}
//catch exception
catch(Exception $e) {
    echo 'Message: ' . $e->getMessage();
}
?>
```

Output:

Message: Value must be 1 or below

[3]



Agenda

- 1 Understanding Exception
- 2 Exceptions in Java
 - Introduction
 - Different Scenarios
 - Catching Exceptions; Syntax, and Example
- 3 Exceptions in PHP
 - Exception without catch
 - Exception with try and catch
- 4 Exceptions in Python
 - Exception without try catch
 - Exception with try catch
- 5 Questions and Discussion



Python: Exception without try catch

Python uses: *try, except, and finally for exception handling*

```
Fatal error: Uncaught exception 'Exception'  
with message 'Value must be 1 or below' in C:\webfolder\test.php:6  
Stack trace: #0 C:\webfolder\test.php(12):  
checkNum(28) #1 {main} thrown in C:\webfolder\test.php on line 6
```

[3]



Agenda

- 1 Understanding Exception
- 2 Exceptions in Java
 - Introduction
 - Different Scenarios
 - Catching Exceptions; Syntax, and Example
- 3 Exceptions in PHP
 - Exception without catch
 - Exception with try and catch
- 4 Exceptions in Python
 - Exception without try catch
 - Exception with try catch
- 5 Questions and Discussion



Python: Exception without try catch

```
#!/usr/bin/python
```



```
try:
    fh = open("testfile", "w")
    try:
        fh.write("This is my test file for exception handling!!")
    finally:
        print "Going to close the file"
        fh.close()
except IOError:
    print "Error: can't find file or read data"
```

[1]



Your Turn: Time to hear from you!



1



¹<https://fensafitters.files.wordpress.com/2013/07/3d095.jpg>

References

-  P.J. Deitel, H.M. Deitel
Java How to program, 10th Edition .
Prentice Hall, 2015.
-  P.J. Deitel, H.M. Deitel
Java How to program, 9th Edition .
Prentice Hall, 2012.
-  Herbert Schildt
The complete reference Java2, 5th Edition .
McGraw-Hill/Osborne, 2002.



References



Java Point

Exception Handling in Java

<http://www.javatpoint.com/exception-handling-in-java> ,
date visited Feb 18, 2017.



Tutorials Point

Java - Exceptions

https://www.tutorialspoint.com/java/java_exceptions.htm ,
date visited Feb 18, 2017.



W3 Schools

PHP Exception Handling

https://www.w3schools.com/php/php_exception.asp , date
visited Feb 18, 2017.



References



Tutorials Point

Python Exceptions Handling

[https:](https://www.tutorialspoint.com/python/python_exceptions.htm)

[//www.tutorialspoint.com/python/python_exceptions.htm](https://www.tutorialspoint.com/python/python_exceptions.htm) ,
date visited Feb 18, 2017.

