

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό – Διάλεξη #14

Διάλεξη #14: Εξαιρέσεις [Exceptions]

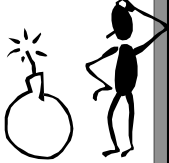
Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 1

Το πρόβλημα

- Τα προγράμματα συχνά περιέχουν λάθη
- Τα λάθη πρέπει να αντιμετωπιστούν
- Ο χειρισμός (αντιμετώπιση) λαθών είναι δύσκολος

Παράδειγμα:

- Διαίρεση με το μηδέν
- δείκτες διανυσμάτων εκτός ορίων



Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 2

Εξαιρέσεις χρόνου-εκτέλεσης [runtime exceptions]


- Μερικές εντολές μπορεί να μην εκτελεστούν επιτυχώς
- Εάν δεν είναι επιτυχείς, τότε **«δημιουργούν μία εξαίρεση»** [throw an exception]
- Οι εξαιρέσεις μπορεί να παραχθούν σε οποιαδήποτε χρονική στιγμή
- Οι εξαιρέσεις διακόπτουν την ροή εκτέλεσης του προγράμματος
- Οι εξαιρέσεις χρησιμοποιούνται για την αναφορά λαθών
- Οι εξαιρέσεις μπορεί να γίνουν αντιληπτές («να συλληφθούν») [can be caught]

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 3

Δημιουργία εξαιρέσεων [throwing exceptions]

- Τα προγράμματα μπορεί να δημιουργήσουν τις δικές τους εξαιρέσεις με σκοπό να αναφέρουν προβλήματα

```
public void remove(int elementNumber)
    throws Exception
{
    if(elementNumber < 0 || elementNumber > count)
    {
        throw new Exception();
    }
    ...
}
```



Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 4

Ο όρος “throws”

- Οι μέθοδοι που δημιουργούν εξαιρέσεις πρέπει να δηλώσουν τις εξαιρέσεις αυτές στην «υπογραφή» τους [signature]

```
public void remove(int elementNumber)
    throws Exception
{
    if(elementNumber < 0 || elementNumber > count)
    {
        throw new Exception();
    }
    ...
}
```

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 5

Η κλάση Exception

- Οι εξαιρέσεις είναι αντικείμενα (στιγμιότυπα της κλάσης `Exception` ή των υποκλάσεων της)

Από τις ιστοσελίδες τεκμηρίωσης:

```
public Exception()
```

Constructs an Exception with no specified detail message.

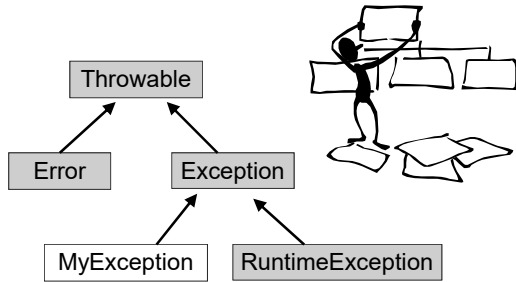
```
public Exception(String s)
```

Constructs an Exception with the specified detail message.
Parameters:
s - the detail message.

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 6

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό – Διάλεξη #14

Η ιεραρχία των εξαιρέσεων



Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 7

Η ιεραρχία των εξαιρέσεων (συνέχεια)

- Η κλάση **Throwable** περιλαμβάνει τα λάθη **[errors]** και τις εξαιρέσεις **[exceptions]**
- Σε αντίθεση με τις εξαιρέσεις, τα λάθη «δεν μπορεί ποτέ να γίνουν αντιληπτά»
- Οι εξαιρέσεις που ανήκουν στην κλάση **RuntimeException** (και τις υποκλάσεις της) δεν χρειάζεται να δηλωθούν σε τμήμα «**throws**» (επόμενο slide)
- Οι εξαιρέσεις που δηλώνονται από τους χρήστες, πρέπει να κληρονομούν από την κλάση **Exception**

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 8

Ελεγχόμενες και μη ελεγχόμενες εξαιρέσεις

[Checked vs. unchecked exceptions]

Ορολογία:

- Οι περισσότερες εξαιρέσεις είναι **ελεγχόμενες** εξαιρέσεις και πρέπει να δηλώνονται σε ένα τμήμα **throws**.
- **Μη ελεγχόμενες** εξαιρέσεις μπορεί να δημιουργηθούν οποιαδήποτε στιγμή και δεν χρειάζεται να έχουν δηλωθεί.
- Οι μη ελεγχόμενες εξαιρέσεις υλοποιούνται στην Java μέσω της κλάσης **RuntimeException**.
- Για τον χειρισμό λαθών, οι χρήστες πρέπει πάντα να χρησιμοποιούν ελεγχόμενες εξαιρέσεις.

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 9

Παροχή περισσότερων πληροφοριών

- Συνήθως, θέλουμε να δώσουμε περισσότερες πληροφορίες σχετικά με το λάθος που προέκυψε

```
public void remove(int elementNumber)
    throws Exception
{
    if(elementNumber < 0 || elementNumber > count)
    {
        throw new Exception("element number out of range");
    }
    ...
}
```

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 10

Η δημιουργία των «δικών μας» εξαιρέσεων



Οι «δικές μας» εξαιρέσεις **[custom exceptions]** δηλώνονται ως υποκλάσεις της κλάσης **Exception**

```
class NumberOutOfRangeException extends Exception
{
    /**
     * Create a new exception with the illegal number
     * as an argument.
     */
    NumberOutOfRangeException(int number)
    {
        super("The number " + number + " is out of range");
    }
}
```

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 11

Η δημιουργία των «δικών μας» εξαιρέσεων

```
public void remove(int elementNumber)
    throws Exception
{
    if(elementNumber < 0 || elementNumber > count)
    {
        throw new NumberOutOfRangeException(elementNumber);
    }
    ...
}
```

Οι «δικές μας» εξαιρέσεις μπορεί

- να αποθηκεύουν επιπρόσθετες πληροφορίες
- να χρησιμοποιηθούν στον χειρισμό άλλων εξαιρέσεων (λεπτομέρειες αργότερα...)

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 12

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό – Διάλεξη #14

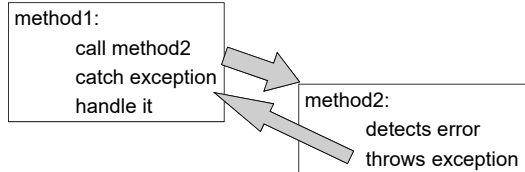
Το πρόβλημα

- Τις περισσότερες φορές δεν θέλουμε μόνο να αναφέρουμε τα προβλήματα. Επιθυμούμε να τα αποκαταστήσουμε!
- Το πρόγραμμα, ως οντότητα, δεν πρέπει να σταματήσει τη λειτουργία του.

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 13

Η «σύλληψη» των εξαιρέσεων [Catching exceptions]


- Οι εξαιρέσεις μπορεί να συλληφθούν.



Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 14

“try” και “catch”

```
...  
int itemNumber = getInputFromUser();  
try  
{  
    database.remove(itemNumber);  
}  
catch (Exception exc)  
{  
    System.out.println("an error occurred: " + exc);  
}  
...
```



Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 15

Χειρισμός διαφόρων τύπων εξαιρέσεων

```
...  
int itemNumber = getInputFromUser();  
try  
{  
    database.remove(itemNumber);  
}  
catch (NumberFormatException exc)  
{  
    System.out.println("an error occurred: " + exc);  
}  
...
```

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 16

“try” και “finally”

```
public void someMethod()  
{  
    ....  
    ....  
    try  
    {  
        // some code  
    }  
    finally  
    {  
        // clean up  
    }  
    ....  
    ....  
}
```

Εκτελείται πάντοτε!

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 17

Παράδειγμα

```
public boolean searchFor(String file, String word)  
    throws StreamException  
{  
    Stream input = null;  
    try  
    {  
        input = new Stream(file);  
        while(!input.eof())  
            if(input.next() == word)  
                return true;  
        return false;  
    }  
    finally  
    {  
        if(input != null)  
            input.close();  
    }  
}
```

Μπορεί να δημιουργήσουν εξαιρεση

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 18

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό – Διάλεξη #14

“finally”

- Το **finally**-τμήμα πάντοτε εκτελείται μετά το **try**-τμήμα, ανεξάρτητα από τον τρόπο με τον οποίο το **try**-τμήμα τερματίστηκε.
- Το **finally**-τμήμα μπορεί να χρησιμοποιηθεί για να εγγυηθεί τον ομαλό τερματισμό μιας σύνθετης λειτουργίας [to clean up after an operation]

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 19