CS112 Lab 04, Feb 11, 14 2010

http://cs-people.bu.edu/deht/cs112_spring11/lab04/

Diane H. Theriault
deh@cs.bu.edu
http://cs-people.bu.edu/deht/
Understanding the Relationship Between Objects

• Your assignment for HW2 is about putting pieces together.

• If you understand the relationship between objects in the program, it will be much easier to think about.

• You do have to implement your own data structures, but you can use the book.
Let’s make a picture

• Your calculator class must have two methods

```
Calculator

evaluate(String)
print()
```

Let’s make a picture

• What data does your calculator need to do its work?

• (Given in assignment)
  – Bag with an iterator
  – Stack
Let’s make a picture

- What is in the Stack? int’s.
- What is in the Bag? “Variable”s

```java
class Variable{
    char name;
    int value
}
```
Error Handling Strategies

• Ignore them (at your peril)
Error Handling Strategies

• Ignore them (at your peril)
• Return true on success, false otherwise
  – How do you know what went wrong?
Error Handling Strategies

• Ignore them (at your peril)
• Return true on success, false otherwise
  – How do you know what went wrong?
• Return error codes
  – How do you keep track of what the codes mean?
Error Handling Strategies

• Ignore them (at your peril)

• Return true on success, false otherwise
  – How do you know what went wrong?

• Return error codes
  – How do you keep track of what the codes mean?

• **Throw exceptions**
  
What is an Exception?

• When something bad happens, your program creates an object and hands it to the JVM

• The JVM takes the object and tries to give it to someone who knows what to do with it.
Familiar Exceptions

• Null pointers
  \((\text{NullPointerException})\)

• Wrong array indexes
  \((\text{ArrayIndexOutOfBoundsException})\)

• Casting between classes
  \((\text{ClassCastException})\)

  Object obj = new String("hello");  
  Integer var = (Integer)(obj);
Syntax

try {
    //your code here
}

catch( ExceptionType myException) {
    //handle this type of error

    throw new DifferentException()
    //if you want to report a different error

}

finally{
    //cleanup
}

Practical Lab

• Runner program does bad things.

• Intercept these bad things and print error messages
You can make your own exceptions

- Define a class that inherits from “Exception”

```java
class MyException extends Exception
{
    MyException(String message)
    {
        super(message);
    }
}
```
Your methods can throw your exceptions

• A method that throws an exception must tell the compiler what types of exceptions it throws.

    static void myMethod() throws MyException
    {
        throw new MyException("uh oh!");
    }
You can catch your own exceptions

```java
try{
    myMethod();
}
catch(MyException exception){
    System.out.println(exception.getMessage());
}
```
You can catch several different types of exceptions in one block

```java
try{
    myMethod();
}
catch(NullPointerException exception){
    System.out.println("this pointer is null");
}
catch(ArrayIndexOutOfBoundsException exception){
    System.out.println("array index out of bounds");
}
```
Practical Lab

• Define your two exception classes (like in the homework)

• Create a function that throws your exceptions in response to some events

• Use try{} catch(...){} to catch your exceptions and do something with them

• No solution will be posted for this exercise