

## Handling Errors

- What happens if someone attempts to add an item to a queue that is full?
- Many times it is the client that knows what the appropriate thing to do is in the case of an error.
- Exceptions provide a method of informing a client that an error has occurred and provides a mechanism for handling the error *cleanly*.
- Many languages provide support for exceptions.

3/14/01

Exceptions

1

---

---

---

---

---

---

---

---

## Java Exception Handling

- An exception occurs, *thrown*, when something goes wrong while a program is running.
- If an exception is not handled, *caught*, the program will terminate.
- When an exception is thrown, the exception is passed to the caller of the method.
- The caller can then handle the exception or allow it to propagate back up the call chain.

3/14/01

Exceptions

2

---

---

---

---

---

---

---

---

## Java Exceptions

- An *exception object* stores information about the cause of an exception. Exceptions must be a subclass of `Throwable`
  - `Throwable` and its subclasses have two constructors, one that takes no arguments, and one that takes a `String` argument that can be used to produce an error message
  - A `Throwable` class contains a snapshot of the state of the program at the time it was created
  - It can also contain a message string that gives more information about the error

3/14/01

Exceptions

3

---

---

---

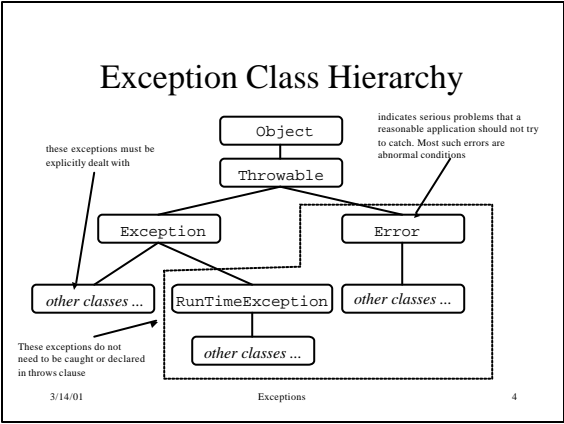
---

---

---

---

---




---

---

---

---

---

---

---

---

- ### Java Exceptions (partial list)
- ActivationException
  - ApplicationException
  - AWTException
  - ClassNotFoundException,
  - DataFormatException
  - GeneralSecurityException
  - IllegalAccessException
  - InstantiationException
  - InterruptedException
  - IOException
  - NoSuchMethodException
  - ParseException
  - ArithmeticException
  - ArrayIndexOutOfBoundsException
  - ArrayStoreException
  - ClassCastException
  - ClassNotFoundException
  - IllegalAccessException
  - IllegalArgumentException
  - IndexOutOfBoundsException
  - NegativeArraySizeException
  - NullPointerException
  - NumberFormatException
  - ...
- 3/14/01                      Exceptions                      5

---

---

---

---

---

---

---

---

- ### What Exceptions are Thrown?
- How do you figure out what exception will be thrown that you must handle?
    - read the documentation for the class you are using
    - read the documentation about the various exception classes
  - I often simply use the methods I want and let the compiler tell me when I missed something!!
- 3/14/01                      Exceptions                      6

---

---

---

---

---

---

---

---

## Try, Catch, and Finally

- A `try` block is wrapped around code that may cause an exception.
- `catch` blocks following the `try` block can contain code to deal with the exception.
- A `finally` block contains code that will be executed whether or not an exception occurs.
- In most cases `try` and `catch` blocks suffice

3/14/01

Exceptions

7

---

---

---

---

---

---

---

---

## Syntax

```
try {  
    statementSequence  
}  
  
catch ( parameter ) {  
    statementSequence  
}  
  
finally {  
    statementSequence  
}
```

3/14/01

Exceptions

8

---

---

---

---

---

---

---

---

## CmdLine

```
public class CmdLine {  
    public static void main( String args[] ) {  
        for ( int i=0; i<args.length; i++ ) {  
            int val;  
  
            try {  
                val = Integer.parseInt( args[i] );  
                System.out.println( val );  
            }  
            catch ( NumberFormatException e ) {  
                System.out.println( "??" );  
            }  
        }  
    }  
}
```

3/14/01

Exceptions

9

---

---

---

---

---

---

---

---

## Java Exception Handling

- An exception occurs, *thrown*, when something goes wrong while a program is running.
- If an exception is not handled, *caught*, the program will terminate.
- When an exception is thrown, the exception is passed to the caller of the method.
- The caller can then handle the exception or allow it to propagate back up the call chain.

3/14/01

Exceptions

10

---

---

---

---

---

---

---

---

## throws

- If a method wishes to pass an exception to a caller, the exception must be listed in the throws clause of the method
  - `public int aMethod() throws NumberFormatException`
- If a method does something that may cause an exception to be thrown
  - the exception must be caught
  - or the exception must be listed in the method's throw clause

3/14/01

Exceptions

11

---

---

---

---

---

---

---

---