Where are We?

Java: read chapters 13, 15, and 17
This should be mostly a review…with the exception of exceptions.

Do exercises: 15.1 a,b and 17.3, due Monday 3/27

Java Exceptions

The term *exception* is shorthand for the phrase "exceptional event."

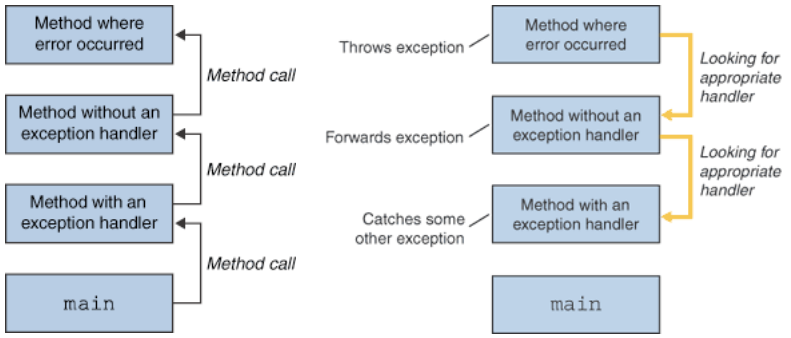
Def**:** An *exception* is an event, which occurs during the execution of a program, that disrupts the normal flow of the program's instructions.

When an error occurs within a method, the method creates an object and hands it off to the runtime system. The object, called an *exception object*, contains information about the error, including its type and the state of the program when the error occurred. Creating an exception object and handing it to the runtime system is called *throwing an exception*.

After a method throws an exception, the runtime system attempts to find something to handle it. The set of possible "somethings" to handle the exception is the ordered list of methods that had been called to get to the method where the error occurred. Handlers *catch exceptions.*

Source: http://java.sun.com/docs/books/tutorial/essential/exceptions/definition.html

Java Exceptions

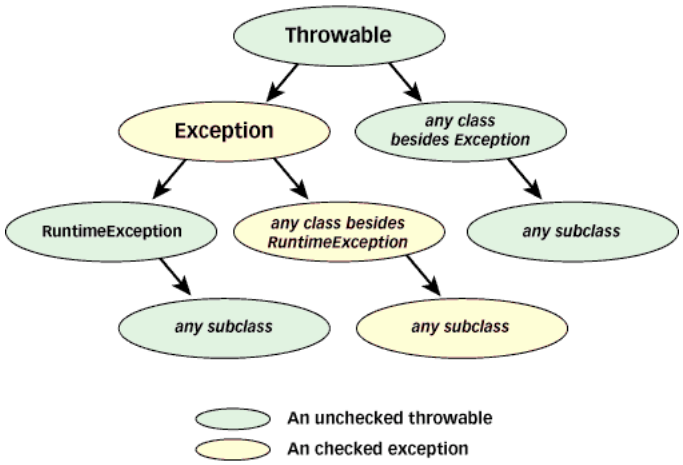


Java Exceptions

```
try {
  // Perform work here
} catch (Exception e) {
  // Log the exception and continue
  System.out.println("Unexpected exception", e);
}


// In some function h()
throw new Exception("optional text here");
```

Java Exceptions



Java Exceptions

The Exception class behaves just like any other class:

```
Class MyException extends Exception {
       MyException (List items) {…};
       void printList() {…}
}
```