Lec 23

Exceptions and Animations in JavaFX
THE END IS NEAR
8AM

A. Love flower
B. Surprise panda
C. Day/Night
D. Birds/Rain
E. Drought
A. Train
B. Pizza Game?
C. Cars + rain
D. Dog heads
E. Solar System
http://i.imgur.com/X17pulB.gif
Exceptions

• 2 types
  – Checked exceptions
  – Runtime (unchecked) exceptions

• Errors
try... catch
	ry {
  // code
} catch(ExceptionType e) {
  // code
}
public void writeArray() {
    FileOutputStream fs = null;
    PrintWriter pw = null;
    int[] a = {0, 1, 2};
    try {
        fs = new FileOutputStream(“out.txt”);
        pw = new PrintWriter(fs, true);
        for(int i = 0; i <= a.length; i++) {
            pw.println(a[i]);
        }
    } catch (Exception e) {
        System.out.println(“problem with file”);
    }
}
A better solution II

```java
public void writeArray() {
    FileOutputStream fs = null;
    PrintWriter pw = null;
    int[] a = {0, 1, 2};
    try {
        fs = new FileOutputStream("out.txt");
        pw = new PrintWriter(fs, true);
        for (int i = 0; i <= a.length; i++) {
            pw.println(a[i]);
        }
    } catch (IOException e) {
        System.out.println("problem with file");
    }
}
```

What happens if:
- We cannot open “out.txt” to write to due to permission issues
  - “problem with file” gets printed
- We can open “out.txt” to write to
  - ArrayIndexOutOfBoundsException
- problem with file gets printed
  - ArrayIndexOutOfBoundsException
public void writeArray() {
    FileOutputStream fs = null;
    PrintWriter pw = null;
    int[] a = {0,1,2};
    try {
        fs = new FileOutputStream("out.txt");
        pw = new PrintWriter(fs, true);
        for(int i = 0; i <= a.length; i++) {
            pw.println(a[i]);
        }
    } catch (IOException e) {
        System.out.println("problem with file");
    } catch (ArrayIndexOutOfBoundsException e) {
        System.out.println("array error");
    }
}
Throwing Exceptions
Methods throwing exceptions

- Methods can cause an exception, and instead of fixing it, just report to whoever called it

```java
public void writeArray() throws IOException, ArrayOutOfBoundsException {
    FileOutputStream fs = null;
    PrintWriter pw = null;
    int[] a = {0,1,2};
    fs = new FileOutputStream("out.txt");
    pw = new PrintWriter(fs, true);
    for(int i = 0; i <= a.length; i++) {
        pw.println(a[i]);
    }
}
```
public class Test {
    public static void main(String[] args) {
        writeArray();
    }

    public static void writeArray() throws IOException {
        FileOutputStream fs = null;
        PrintWriter pw = null;
        int[] a = {0, 1, 2};
        fs = new FileOutputStream("out.txt");
        pw = new PrintWriter(fs, true);
        for (int i = 0; i < a.length; i++) {
            pw.println(a[i]);
        }
    }
}

What happens if:
- We cannot open “out.txt” to write to due to permission issues
- We can open “out.txt” to write to

A) – exit gracefully, nothing printed
    - out.txt gets expected output
B) – program exits with run time error
    - program exits with run time error
C) – exit gracefully, nothing printed
    - exit gracefully, nothing printed
D) – Compiler error
E) – None of the above
Solution

```java
public class Test {
    public static void main(String[] args) {
        try {
            writeArray();
        } catch (IOException e) {
            System.out.println("problem");
        }
    }

    public static void writeArray() throws IOException {
        FileOutputStream fs = null;
        PrintWriter pw = null;
        int[] a = {0, 1, 2};
        fs = new FileOutputStream("out.txt");
        pw = new PrintWriter(fs, true);
        for (int i = 0; i < a.length; i++) {
            pw.println(a[i]);
        }
    }
}
```

What happens if:
- We cannot open “out.txt” to write to due to permission issues
- We can open “out.txt” to write to
A (bad) solution

```java
public class Test {
    public static void main(String[] args) throws IOException {
        writeArray();
    }

    public static void writeArray() throws IOException {
        FileOutputStream fs = null;
        PrintWriter pw = null;
        int[] a = {0, 1, 2};
        fs = new FileOutputStream("out.txt");
        pw = new PrintWriter(fs, true);
        for(int i = 0; i < a.length; i++) {
            pw.println(a[i]);
        }
    }
}
```

What happens if:
- We cannot open “out.txt” to write to due to permission issues
- We can open “out.txt” to write to
finally

try {
    statements;
}

catch (TheException e) {
    handle exception;
}

finally {
    finalStatements;
}
Which statements execute?

```java
try {
    statement1;
    statement2;  //causes Exception1 to occur
    statement3;
}
catch (Exception1 e) {
    statement4;
}
finally {
    statement5;
}
statement6;
```

A) 1,2,3,4,5,6
B) 1,2,4,5,6
C) 1,2,5,6
D) 1,2,4,6
E) 1,2,6
Animations

- Circle
- Rectangle
- Line
- ...

- Have access to event handlers
How would you design

- On mouse click circle is created and moves to the right
- Multiple circles can be created
- Label is update, counting number of mouse clicks

- What Layout manager would you use?
- What classes? What do they extend, implement?
- What are your EventHandlers?