

Slide 1

Administrivia

- (How was Tuesday's class? Did you do any code examples?)
- Project description on Web at last, plus first homework (design phase now, rest to be there later today or tomorrow). Mostly this assignment is about you planning your game and figuring out a bit about Eclipse. Design (description of game) due Tuesday; code due Thursday.
- Quiz 1 rescheduled for Tuesday. Should we move to Thursday? (Consensus: Yes.)

Slide 2

Packages and Importing

- Library classes grouped into "packages" — e.g., `java.util`, `java.net`.
- For classes in `java.lang` and "default package", reference using their names only. For other classes, can use full name or `import`. (`import` looks like `#include`, but works differently.)
- You can define your own packages. Convention is to use your e-mail/Web address, in reverse order (e.g., Dr. Lewis's framework is `edu.trinity.cs.gamecore`). For your game, I'm recommending `edu.trinity.cs.yourusername`. You could add `pad2game` if you wanted to.

Exceptions

- We'll talk more about this later, but for now:
- Exceptions are meant as a flexible and robust way to deal with errors, better in some ways than other ways.
- Typical usage is like this (don't worry about details, just overall idea):

Slide 3

```
try {
    BufferedReader rdr =
        new BufferedReader(new FileReader(args[0]));
    /* other code using rdr */
}
catch (FileNotFoundException e) {
    System.err.println("file " + args[0] + " not found");
}
catch (IOException e) {
    e.printStackTrace();
}
```

Polymorphism Example

- Let's write a bit of code illustrating the use of inheritance, types, etc. ...

Slide 4

Minute Essay

- Have you tried writing some simple Java programs yet?
- Do you feel like you're getting the basic ideas? if not, are there specific things you want to hear more about?

Slide 5