Lists, Loop Idioms, Generics, and the Null Keyword

Getter method for lists

```java
public List<Book> getContents()
    {
        return this.contents;
    }
```

Idiom - common pattern or expression

For each loop over list

```java
for (Book book : this.contents)
```

Break; - terminates loop

Generics

```java
ClassName<...>
// you can write anything in place of ...
// ... is an unspecified type name
Box<String> box1 = new Box<String>("surprise");
Box<Integer> box2 = new Box<Integer>(42);
```

Errors

compile time error - invalid syntax (trying to use the wrong type for something)

run time error - invalid parameters (out of bounds, etc)

Null

- use null to initialize objects that don't need a value
- any variable that refers to an object can be initialized to null

  NullPointerException causes:

- a field or variable isnt being initialized
- the result of a method being used is null
- you are using null as a parameter instead of an object

```java
Pixel pix = null;
pix.setRed(255); // This was a cause a NullPointerException
```