Harry Potter and the Philosopher's Stone
J. K. Rowling
2000

Where the Wild Things Are
Maurice Sendak
1970

Where the Wild Things Are
Maurice Sendak
1970

Oh, the Places You'll Go!
Dr. Seuss
1990
Announcements

• Next time - Review for Final
• More on Recursion
• Complexity and Abstraction
• Recursive Art Winners!
Classwork today

- Library of books
  - Title
  - Author
  - Date pub.
- Search for books
- Sort the books found

<table>
<thead>
<tr>
<th>Title</th>
<th>Author</th>
<th>Date pub.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Harry Potter and the Philosopher's Stone</td>
<td>J. K. Rowling</td>
<td>2000</td>
</tr>
<tr>
<td>Where the Wild Things Are</td>
<td>Maurice Sendak</td>
<td>1970</td>
</tr>
<tr>
<td>Oh, the Places You'll Go!</td>
<td>Dr. Seuss</td>
<td>1990</td>
</tr>
</tbody>
</table>
Result of Search - Katz

We have 1 copy of Counting Kisses: A Kiss and Read Book by Karen Katz published in 2000
We have 1 copy of Daddy and Me by Karen Katz published in 2003
We have 1 copy of Where Is Baby's Belly Button? by Karen Katz published in 2002

Found 3 books!
Want to Sort Answer

- Several ways
  - By Title first
  - By Author first
  - By Counts (number of copies) first
Interface Types

- Makes code more general and more reusable
- Express common operations
Example for Today

• public interface Comparator<T>
  – Has a compare method
• Implement compare by Titles

```java
package yourwork;
import java.util.Comparator;

public class TitleComparator implements Comparator<Book> {
    public int compare(Book left, Book right) {
        return left.getTitle().compareTo(right.getTitle());
    }
}
```
You’ll implement

- Comparator for Author first, etc…
- Comparator for Count of number of copies first, etc..
- See handout