Interfaces

Looking back:
- Data types & expressions
- Classes & objects, encapsulation
- Algorithms, recursive & iterative
- Data Structures
  - Contiguous (Java arrays)
  - Pointer-based (various Java collections + build our own)

But first: need to understand interfaces
Interface — a data type without an implementation

```java
public interface List {
    public boolean add(Object x);
    " " contains (Object x);
    " " remove (Object x);
    public int size();
    public void clear();
    public Iterator iterator();
}

an interface too!

public interface Iterator {
    public boolean hasNext(); // are there more items?
    public Object next(); // returns the next item
    public void remove(); // removes item that next returned
}
```
addToShoppingCart(List cart, Item item) {  
cart.add(item);  
  Iterator it = cart.iterator();  
  double total = 0;

  while (it.hasNext()) {  
    Item x = (Item) it.next();  
    total = total + x.price();
  }
}
addToShoppingCart(List cart, Object item)

    cart.add(item);

    double total = 0;
    for (Object x : cart)
        total = total + ((Item) x).price();

}
public class LinkedList implements List {
	null;

} // obliged to provide implementations of all the List methods

new List();

l = new LinkedList();
addToShoppingCart(l, foo);