Copyright 2003 Bill Campbell and Ethan Bolker

* A ShoppingCart keeps track of a customer's purchases.

* @see EStore
* @version 4

public class ShoppingCart
{
/// replace these two fields by a single ArrayList
private int count; // number of Items in this ShoppingCart
private int cost;  // total cost of Items in this ShoppingCart

/**
* Construct a new empty ShoppingCart.
*/

public ShoppingCart()
{
    count = 0;
    cost  = 0;
}

/**
* Add an Item to this ShoppingCart.
* @param item the Item to add.
*/

public void addItem( Item item )
{
    count++; // Java idiom: a += b means a = a + b
    cost += item.getCost(); // Java idiom: a += b means a = a + b
}

/**
* Return an Item from this ShoppingCart.
* @param item the Item to return.
*/

public void returnItem( Item item )
{
    /*
     * look through the list looking for Item
     * remove it if it's there
     */
}

/**
* What happens when this ShoppingCart is asked how many
* Items it contains.
* @return the number of items in this ShoppingCart.
*/

public int getCount()
{
    return count;
}

/**
* What happens when this ShoppingCart is asked the total
* cost of the Items it contains.
* @return the total cost of the items in this ShoppingCart.
*/

public int getCost()
{
    return cost;
}

/**
* Write the contents of this ShoppingCart to a Terminal.
* @param t the Terminal to use for output.
*/

public void showContents( Terminal t)
{
    /*
     * work to do here ...
     */
    t.println("  [sorry, can't yet print ShoppingCart contents]");
}
}