COMP 212 Spring 2025
Homework 06

1 Shopping Cart

Your goal is to implement a simple shopping cart, using the same ideas as last Thursday's lecture: we represent the states of an interactive application by the constructors of a `datatype`

An interaction with your shopping cart should look roughly like this:

```
Please enter your name:
[you type] Dan
Hi, Dan What would you like to buy?
  apples  $1/pound
  bananas $2/bunch
  cookies $2/box
Or say 'checkout' to check out.
[you type] apples
Hi, Dan What would you like to buy?
  apples  $1/pound
  bananas $2/bunch
  cookies $2/box
Or say 'checkout' to check out.
[you type] bananas
Hi, Dan What would you like to buy?
  apples  $1/pound
  bananas $2/bunch
  cookies $2/box
Or say 'checkout' to check out.
[you type] apples
Hi, Dan What would you like to buy?
  apples  $1/pound
  bananas $2/bunch
  cookies $2/box
Or say 'checkout' to check out.
[you type] checkout
Hi, Dan
Your cart contains apples,bananas,apples.
```

```
I will charge you $4.
Type 'pay' to pay.
[you type] pay
Please enter your name:
...
```

The key features are:

- At the start, the application should ask for the user's name, which should be displayed while they are shopping.
- The application should repeatedly ask the user to select something to buy, which they can choose by typing the name of the product.
- Each product should have a price.
- Typing `checkout` should take the user to a screen that lists the items they have selected along with the total price.
- Then typing `pay` should take the user back to the beginning (with the idea that it has paid for the order).

You can choose what products are for sale, how much they cost, and exactly how things are displayed. For simplicity, the list of products and prices can be fixed/hard-coded into the code, though in reality this would be pulled from a database that is updated as people buy things.

Task 1.1 (10 pts). Choose constructors for the datatype `model` to represent each state of the application, and pick arguments to the constructors that store the information associated with each state. For example, during the shopping phase, you will need to choose a representation for the shopping cart.

Task 1.2 (10 pts). Write a function `view : model -> string` that displays the model as a string.

Task 1.3 (10 pts). Write a function `respond : model * string -> model` that updates the model state based on user input.

See the instructions in `hw06.sml` for how to run your application using the `run()` function.