# The Shopping List

## Lelani and Jamie

Visual Studio Code — Wed 15:54

app.rb - app - Visual Studio Code

dit Selection View Go Debug Terminal Help

list.rb | app.rb | README.md | items.rb | Gemfile

```ruby
require_relative './classes/items'
require_relative './classes/list'
require 'tty-prompt'

# Creates new object
ordered_list = List.new

# Main Loop - Asks customer for options
loop do
    puts "Press (a)dd to add an item, or \nPress (p)review to view your current list, or \nPress (q)uit to exit once finished"
    answer = gets.strip.downcase

    # Categories array
    categories = %w(fruit/veggies chilled/frozen meat/poultry bakery other)

    case answer
    # When input is (a)
    when "a"
        puts "What item would you like to add to your shopping list?"
        item_name = gets.strip.downcase
        prompt = TTY::Prompt.new
        category = prompt.select('Choose your category', categories, help: '(Use arrow keys and Enter to finish)')
        item = Item.new(item_name, category)
        puts "How many #{item_name}'s would you like?"
        quantity = gets.strip.to_i
        ordered_list.add_item(item, quantity)
    # When input is (p)
    when "p"
        ordered_list.preview
    # When input is (q)
    when "q"
        ordered_list.print_final_list(categories)
        break
    else
        puts "Invalid option"
    end
end
```

0 0 — Ln 5, Col 1 Spaces: 4 UTF-8

# Demo

# Challenges

We did come across a few challenges.

- Incorporating the gem was our biggest challenge, the first gem we tried to use was unclear and we ended up going with a different gem. It was then unclear if our gems information was returning our input.
- Our second biggest challenge was implementing the class. Especially when trying to keep track of and come up with new variable names as well as calling on methods.

# What we enjoyed

I enjoyed pair programming, I loved the fact that when we actually started coding we were able to shoot ideas and suggestions to each other. It made the process easier and faster.