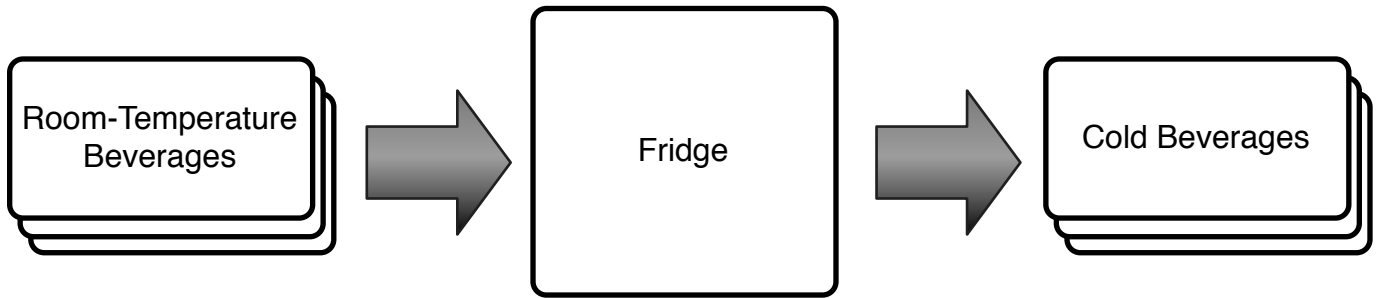


Passive-Aggressive Note #1: How to Maintain a Supply of Cold Beverages



```
queue = [  
  'peach tea'  
  'peach tea'  
  'lemon tea'  
  'lemon tea'  
  'raspberry tea'  
  'raspberry tea'  
]  
  
# This module exposes a "restock" function which takes one or more  
# beverages and adds them to the queue:  
#  
# > queue.length  
# 6  
# > restock('peach tea', 'peach tea')  
# > queue.length  
# 8  
#  
# The "restock" function returns an object with a "take" method which  
# returns a cold beverage. It's common to restock a warm beverage in  
# order to take a cold beverage of the same type:  
#  
# > restock('lemon tea').take('lemon tea')  
# 'lemon tea'  
#  
# Restocking more than one beverage is encouraged, and it's acceptable  
# to take a beverage of a different kind from those restocked:  
#  
# > restock('peach tea', 'lemon tea').take('raspberry tea')  
# 'raspberry tea'  
#  
# Trying to take a beverage without first restocking raises an error:  
#  
# > restock().take('lemon tea')  
# Error  
#  
# Attempting to take a beverage which is not in the fridge raises a  
# "beverage not found" error:  
#  
# > restock('lemon tea').take('orange juice')  
# Error  
restock = (beverages...) ->  
  throw new Error 'no beverages provided' unless beverages.length  
  queue = [queue..., beverages...]  
  take: (beverage) ->  
    idx = queue.indexOf beverage  
    throw new Error 'beverage not found' unless ~idx  
    queue.splice(idx, 1)[0]  
  
module.exports = {restock}
```