Mini Project Week 3

So far we've been using one-dimensional lists of data, however, this won't work for orders. We need to store more information such as the customer's name, address and phone number, as well as the status of the order, the courier etc. To solve this we'll use a two-dimensional data structure, a `dictionary`. We won't be able to read/write this structure to text file anymore, but we'll fix this later. For now we'll also skip adding products to the order.

We'll also write a unit-test that covers the update order status functionality.

Goals

As a user I want to:

- create a product, courier, or order and add it to a list
- view all products, couriers, or orders
- update the status of an order
- persist my data
- *STRETCH* update or delete a product, order, or courier

Spec

- A `product` should just be a `string` containing its name, i.e: `"Coke Zero"`
- A list of `products` should be a list of `strings`, i.e: `["Coke Zero"]`
- A `courier` should just be a `string` containing its name, i.e: `"John"`
- A list of `couriers` should be a list of `strings`, i.e: `["John"]`
- An `order` should be a `dict`, i.e:

```
{
  "customer_name": "John",
  "customer_address": "Unit 2, 12 Main Street, LONDON, WH1 2ER",
  "customer_phone": "0789887334",
  "courier": 2,
  "status": "preparing"
}
```

- A list of `orders` should be a list of `dicts`, i.e: `[{...},{...}]`
- Data should be persisted to a `.txt` file on a new line for each `courier` or `product`, ie:

```
John
Claire
```

Pseudo Code

- START APP
- LOAD `COURIERS` AND `PRODUCTS` FROM `TXT` FILE
- SHOW LIST OF OPTIONS TO USER AND ACCEPT NUMERICAL INPUT
- IF USER ENTERS `0` THEN SAVE APP STATE TO `TXT` FILES AND EXIT APP
- IF USER ENTERS `1` THEN SHOW `PRODUCT` MENU
    - IF USER ENTERS `0` RETURN TO MAIN MENU
    - IF USER ENTERS `1` PRINT OUT `PRODUCTS` TO SCREEN
    - IF USER ENTER `2` CREATE NEW `PRODUCT`
    - ASK USER FOR THE `NAME` OF THE `PRODUCT`
    - APPEND THIS TO THE LIST OF `PRODUCTS`
    - *STRETCH* IF USER ENTERS `3` UPDATE `PRODUCT`
    - ASK USER TO SELECT A `PRODUCT` TO UPDATE
    - ASK USER FOR NEW `NAME` OF `PRODUCT`
    - REPLACE `PRODUCT` AT SELECTED `IDX` WITH NEW `NAME`
    - *STRETCH* IF USER ENTERS `4` DELETE `PRODUCT`
    - ASK USER TO SELECT A `PRODUCT` TO DELETE
    - REMOVE THIS ITEM FROM THE `PRODUCTS` LIST
- IF USER ENTERS `2` THEN SHOW `COURIER` MENU
    - IF USER ENTERS `0` RETURN TO MAIN MENU
    - IF USER ENTERS `1` PRINT OUT `COURIERS` TO SCREEN
    - IF USER ENTER `2` CREATE NEW `COURIER`
    - ASK USER FOR THE `NAME` OF THE `COURIER`
    - APPEND THIS TO THE LIST OF `COURIERS`
    - *STRETCH* IF USER ENTERS `3` UPDATE `COURIER`
    - ASK USER TO SELECT A `COURIER` TO UPDATE OR `0` TO CANCEL
    - ASK USER FOR NEW `NAME` OF `COURIER`
    - REPLACE `COURIER` AT SELECTED `IDX` WITH NEW `NAME`
    - *STRETCH* IF USER ENTERS `4` DELETE `COURIER`

    - ASK USER TO SELECT A `COURIER` TO DELETE OR `0` TO CANCEL
    - REMOVE THIS ITEM FROM THE `COURIERS` LIST
- IF USER ENTERS `3` THEN SHOW `ORDER` MENU
    - IF USER ENTERS `0` RETURN TO MAIN MENU
    - IF USER ENTERS `1` PRINT OUT `ORDERS` TO SCREEN
    - IF USER ENTER `2` CREATE NEW `ORDER`
    - ASK USER FOR THE `NAME` OF THE `CUSTOMER`
    - ASK USER FOR THE `ADDRESS` OF THE `CUSTOMER`
    - ASK USER FOR THE `PHONE` OF THE `CUSTOMER`
    - AKS THE USER TO SELECT A `COURIER` FROM THE LIST
    - SET THE DEFAULT `ORDER STATUS` TO BE `PREPARING`
    - APPEND THE NEW ORDER TO THE LIST OF `ORDERS`
    - IF USER ENTERS `3` UPDATE `ORDER STATUS`
    - ASK USER TO SELECT AND `ORDER` TO UPDATE OR `0` TO CANCEL
    - ASK USER TO SELECT A NEW `STATUS`
    - UPDATE THE `ORDER`
    - *STRETCH* IF USER ENTERS `4` UPDATE `ORDER`
    - ASK USER TO SELECT AN `ORDER` TO UPDATE OR `0` TO CANCEL
    - FOR EACH `ORDER` PROPERTY
        - ASK USER FOR UPDATED DATA OR LEAVE BLANK TO SKIP
        - UPDATE THE `ORDER` PROPERTY IF NOT BLANK
    - *STRETCH* IF USER ENTERS `5` DELETE `ORDER`
    - ASK USER TO SELECT AN `ORDER` TO DELETE OR `0` TO CANCEL
    - REMOVE THIS ITEM FROM THE `ORDERS` LIST