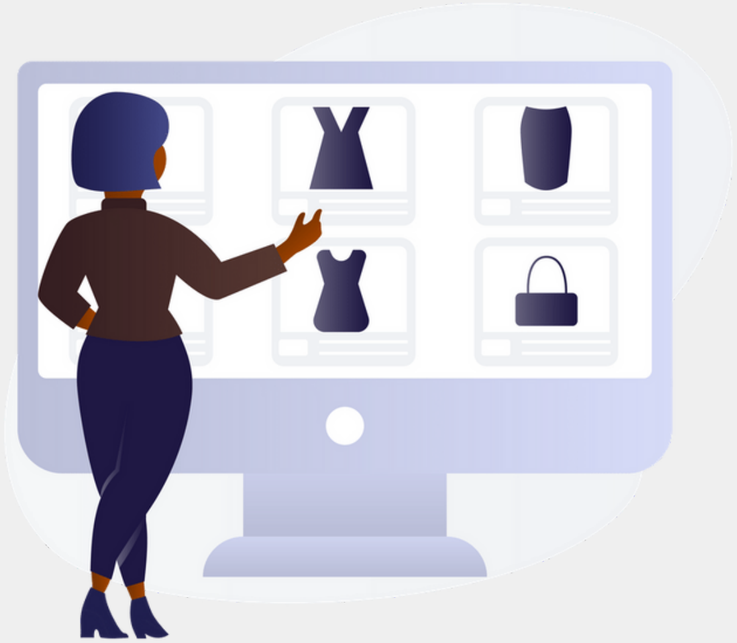
Assignment #3: Data Structures part 2
Act on your Information -- OUTPUT

HW3: GIVE CASCADING DISCOUNTS TO EACH GROUP OF CUSTOMERS, BY AMOUNT, AND ADD DELIVERY DISCOUNT BASED ON NUMBER OF DELIVERIES.

BREAK CUSTOMERS INTO 5 GROUPS BASED ON AMOUNT, AND 3 GROUPS BASED ON NUM DELIVERIES.

CREATE A NEW DICTIONARY REFLECTING THIS, USING THE LABEL AS KEY, THEN THE DISCOUNTS AS A SUB-DICTIONARY USING LABELS 'SPENDING DISCOUNT', 'GOOD CUSTOMER DISCOUNT'.



IN YOUR SUBMISSION, WRITE COMMENTS EXPLAINING YOUR GROUPINGS.

PRINT OUT YOUR RESULTS USING THE FORMAT:
00448329-8DDC-403E-A88A-0ED20E318695 --> {'SPENDING DISCOUNT': 25, 'GOOD CUSTOMER DISCOUNT': 150}

ZIP FILE AND TURN IN USING E-LEARNING.

Assignment #3: Data Structures part 2
Helping Hand

Sort Dictionary: dictSortedByNumDeliveries = dict(sorted(dsDict1.items(), key=lambda item: item[1].get('Num Deliveries'))

```
discountCascadeDeliveries = [ 10, 50, 100, 150, 200]
deliveryDemarcations = [10, 50, 100, 200, 500]
def checkDeliveryDiscount(val):
    res = 4
    ndx = 0
    for demarcation in deliveryDemarcations:
        if val <= demarcation:
            res = ndx
            break
        ndx += 1
    return res
for key, val in dsDict1.items():
    deliveryDiscount = checkDeliveryDiscount(val.get("Num Deliveries")
```