

CIS 122 Winter 2015 Project 3 Lists

Due Friday Feb. 13th, 5 PM

Briefly

Submit several Python 3 programs
Your programs are worth a total of 50 points

Extra work gets extra credit

Project3a.py Display a static list 5 points

Create a list with several entries
Display the list contents, line by line using a
for item in my_list style of loop.

Project3b.py Track income tax data 10 points

Using your Project 2c code as a starting point, create
3 lists: name_list, income_list, tax_due_list
They will start as empty lists, then you append data to
each list.

After you have computed the total tax and average tax,
display each taxpayer's name, income and tax
one line for each taxpayer.

For 3 points **extra credit**, make the data line up nicely
somewhat like this

Income	Tax Due	Name
5,000	0.00	John
10,000	100.00	Annette
300,000	4,875.00	Wanda

Project 3c. List of Bird Species 10 points

Local birdwatchers need your help; they want a list
of local birds observed, but no duplicates.

Ask for a bird name; if the bird is not in the bird_names
list, append it to the list.

If it is already in the list, do not append it.
Instead, print something like "Already in the list."

When you finish collecting the data, print each bird
species in the bird_names list.

Project 3d. Shopping Cart for Valentines 25 points

Shope-Mart is holding a Valentine's Day Special Sale.

It offers a list of Valentine products: and prices

Euphoria Truffles	8.95 each
Valentine's Card	3.75 each
A Dozen Roses	27.95 per dozen
Balloons	12.95 each
Champagne	22.70 each

Your "shopper" will need to see the products and prices,
then make a choice of what to add to his or her
"shopping cart". Finally, the shopper will choose to
either "pay" or cancel the order.

If they decide to pay, total up what they owe:
prices times quantities of each product in the cart.

Extra Credit

- +5 Allow multiple shoppers, each creating
a shopping cart.
- +5 Keep an "inventory" and note changes to
inventory based on shopper's actions.
- +2 Keep a "sale price" for each item, lower than
the standard price. Display it, and calculate
amount owed using the sale prices.

Details

project3a.py Display a static list 5 points:

Create a list of items:

Authors, TV Shows, Cars, UO Majors, Coffees -
you decide.

Put your items into a list of at least 8 items.

Display the list -- print each item in your list on a
separate line.

Here's some code that is similar to this part of your project.
You can choose your topic, but not Wildlife.

```
# lists0.py

# -----
# Static lists can be handy
# Assign a series of values, separated
# by commas between the square brackets
# -----

wildlife = ['Deer', 'Goose', 'Cougar', 'Raccoon', 'Beaver',
            'Opposum', 'Squirrel', 'Nutria']

print('Some Eugene area wildlife \n')
for species in wildlife:
    print(species)
#end for

Some Eugene area wildlife

Deer
Goose
Cougar
Raccoon
Beaver
Opposum
Squirrel
Nutria
```

project3b.py Remember names, incomes, taxes 10 points:

Create 3 empty lists (you can choose your own names for the lists)

```
name_list = [ ]
income_list = [ ]
tax_due_list = [ ]
```

Copy your project 2c income tax Python, and add these lists.
As your program gathers name and income and computes tax due,
you will append that data to the appropriate list.

After your program prints out the total taxes for the day, print
the name, income and tax due for each resident of the fine city of
Plutonium.

Use the sum(my_list) function to add up total tax due; print it.
Use the len(my_list) function to find the total number of citizens
who came by the pay their taxes; print that number too.

project3c.py Make a list of bird species 10 points:

If you have a list like this

```
my_list = ['Mitt', 'Mark', 'Molly']
You can check whether a name is in the list
```

```
name = 'Margaret'
if name in my_list:
    print(name, 'is in our list of winners!')
```

Your job is to start with an empty list of bird names, then use a
while loop to get bird species.

If the species is already **in** the list, just print a "already in the list"
message.

If not, **append** the bird name to the list of bird names.

project 3d Shoppe-Mart Valentine's Specials 25 points

Here is a possible "conversation" between a shopper and your program. Display a menu of choices

```
Shoppe-Mart Valentine Specials
Item Product..... Price
1 Chocolate Truffles 8.95
2 Valentine's Card 3.75
3 A Dozen Roses 27.95
4 Balloons 12.95
5 Champagne 33.70
```

Ask user's choice, then display new shopping cart contents

```
Your choice (1-5) - 0 to end shopping 4
You chose Balloons
```

```
Shopping Cart Contents:
Item Product Price Qty Subtotal
1 Balloons 12.95 1 12.95
```

Ask whether to **continue shopping, check out** and pay for the shopping cart, or **cancel** and leave without buying anything.

For those who want to pay, display shopping cart contents and ask for "payment" after displaying total value of shopping cart.

Static lists are nice for the product names and prices.

You'll decide how to represent a "shopping cart" -- a list of product numbers, or a list of quantities, indexed by product number.

Hints

Creating functions to do some common tasks can keep this task under control.

Start with a simple loop; make sure you can start the shopping loop and stop it with either a purchase or a cancel.

Gradually add more products. You could start with just two, then add some as you get your program working.