

HOMEWORK 7 - Modules

REGULAR VERSION

PART I

Inside your the **week7** directory, create a directory called **hw7_1**. Inside the **hw7_1** directory, create a file called `main.py` and a directory called **animals** which holds 2 python files (modules) - `dog.py` and `labrador.py`.

- Each of these modules should contain a single class named after the module itself (so the `dog.py` file should contain a `Dog` class and the `labrador.py` file should have a `Labrador` class).
- The `Dog` class should have these 2 properties:
 - Name
 - Size ('small', 'medium', or 'large')
- The `Dog` class should have these 2 methods:
 - Bark - return 'woof, woof! I'm a ____ sized dog'
 - Greet - return 'My name is ____ and I LOVE YOU!'
- The `Labrador` class should extend the `Dog` class and override its Greet method so that it returns 'My name is ____ and I LOVE YOU and LOVE to Play!'

In the **week7** directory, create a `main.py` file that creates a `Dog` object and a `Labrador` object. Call the objects' Bark and Greet methods to prove that your code works.

Before submitting, please make sure all executable code is encapsulated in a `main()` function as described in [this week's notes](#).

PART II

Inside your **week7** directory, create a file called `guessing_game.py`. In this file, create a guessing game program that:

- Prompts user for a lowest possible integer to guess
- Prompts the user for a highest possible integer to guess
- Generates a random integer between the lowest and highest possible integers
- Allows the user a maximum of 5 tries to guess that random integer
- If the user guesses before 5 tries, say "CONGRATULATIONS, THE NUMBER WAS ____!"
- If the user fails to guess it after 5 tries, just print out the integer with 'WA..WAAAA....THE NUMBER WAS ____'

Before submitting, please make sure all executable code is encapsulated in a `main()` function

as described in [this week's notes](#).

PART III

Inside your **week7** directory, create a file called `shopper.py`. This program:

- Prompts the user for their groceries one-by-one (5 items)
- Adds each grocery as a string to a list called `shopping_cart`
- If the grocery `'beer'`, `'wine'`, or `'liquor'` is found in `shopping_cart`, then:
 - Prompt the user for their **birthday (not their age)**
 - Check if they are 21 years of age or older
 - If they are too young, chastise them (“You are too young, son!”) and remove the offending groceries from the list
 - If they are of age, thank them - “Thanks and Enjoy!”

Before submitting, please make sure all executable code is encapsulated in a `main()` function as described in [this week's notes](#).

PART IV

Week 8 is very important and the class moves very fast so you will strongly benefit by reading ahead.

Please take 20 minutes to start pre-reading [the notes](#) for this coming week.