**INFO 6350 Fall 2017**

**Swift Practice**

Using Swift playground and / or the command line for macos (open xcode, create a new xcode project, macOS, command line tool), practice the following exercises:

Exercise: Swift Variables

1. Create a variable with a name of your choosing that has a value of 15
2. Create a second variable with a name of your choosing that has a value of 5
3. Create a third variable that stores that sum of the first and second variables
4. Create a fourth variable that stores the result of a multiplication operation
5. Create a fifth variable that stores the result of the first variable minus the second variable
6. Create a sixth variable that stores the result of the first variable divided by the second variable
7. Create a variable of type String with a value of “Hello”
8. Create another variable of type String with a value of “World”
9. Create a third variable to store the result of concatenating the first two strings

Exercise: Swift Functions

1. Create a function named add that takes two parameters of type double and returns the sum of the two numbers
2. Create a function named subtract that takes two parameters of type int and returns the difference of the two numbers
3. Create a function name multiply that takes two parameters of type Float and returns the product of the two numbers
4. Make sure that the three functions created above work by testing them

Exercise: Swift Arrays

1. Declare an empty array of type String
2. Declare an explicit array of type Double and initialize it with 4 values
3. Declare an array and initialize it with 5 values of any type using inference
4. Use append 3 times in each array
5. Use removeAt in each array. Use random to randomly remove an element in the last array
6. Use removeAll in one array

Exercise: Swift Loops

1. Create an empty array of type Int called oddNumbers
2. Using a standard for-in loop add all odd numbers less than or equal to 100 to the oddNumbers array
3. Create a second array called sums of type int
4. Using a for-in loop, iterate through oddNumbers array and add the current iteration value + 5 to the sums array
5. Using a repeat while loop, iterate through the sums array and print “The sum is: x” where x is the current value of the iteration

Exercise: Swift Conditions

1. You are given a fridge that knows when your food is going bad. You know that milk spoils after 21 days and eggs after 10 days. Given milkAge and eggsAge, write a function to determine if you should throw the milk, the eggs or both away or not. If you can keep both, print “ you can still use your milk and eggs”. If you should throw away the milk, print “ you should throw away the milk”. Similarly for the eggs.
2. Write a function that takes in three variables “first, “second” and “third” that checks if at least two variables have the same value. If true, print “two values are at least identical” else print “the values are different”.

Exercise: Swift Dictionaries

1. Create an array of dictionaries in which each dictionary in the array contains the keys “firstName” and “lastName”. Create an array with a name of your choosing that contains only the values for “firstName” in each dictionary.
2. Using the array of dictionaries created previously, this time create an array that contains the values for “firstName” and “lastName” in each dictionary separated by a delimiter of your choose.

Exercise: Swift Tuple - Enum

1. Create an enum that describes different coins values.
2. Create a tuple with a name of your choosing that has (amount, coinType).
3. Print the total value of the coins in the array.

Example:

Var money: [(Int, CoinType)] = [(5, .quarter]),

(6, .dime),

(2, .penny)]

CoinType being the enum that you created. In your enum, let’s say you gave quarter a value of 25, dime a value of 10 and penny a value of 1.

Total should be: 5\*25 + 6\*10 + 2\*1 = 187