

## CS103 – Fall 2025 – **Lab06**

### Overview

- To practice using `while` loops, tuples, and string handling in Python.
- To reinforce counting, user input, and conditional logic.
- To build problem-solving skills with real-world style tasks.

### General Instructions

1. In today's lab, you are required to complete all required exercises during class to receive attendance credit.
2. You are welcome to:
  - Work together with classmates.
  - Search online for help or documentation.
  - Ask the TA for guidance if you are stuck.
3. To receive credit, you must finish and show your solutions to the TA before leaving lab.
4. There are also extra (optional) exercises at the end for those who want to challenge themselves. These are not required for attendance credit but are recommended for practice.

### Exercise Instructions

- 1) Make a folder **Lab6** inside your **cs103fa25** folder.
- 2) Create a new notebook inside your Lab6 folder (`lab06.ipynb`).

## Required Exercises

### EXERCISE 1:

Write a function **mOfThree** that takes an int  $n$  and prints all integers between 1 and  $n$  that are multiples of 3. Use a **while loop** to solve this exercise.

#### Sample Input

```
n = 14
```

#### Expected Output

```
3 6 9 12
```

### EXERCISE 2:

Write a function **tupleCheck** that takes a tuple  $t$  and prints the number of even and odd numbers. Assume the tuple only includes integers.

#### Sample Input

```
t = (1, 2, 3, 4, 5, 6, 7, 8, 9)
```

#### Expected Output

```
Number of even numbers : 4
```

```
Number of odd numbers : 5
```

**EXERCISE 3:**

Write a function **sConcatenate** that does not receive any input.

- The function repeatedly asks the user to enter strings.
- If the user enters "done", the program stops asking and prints the concatenation of all strings entered with a space in between each one.

**1. Sample Run**

```
Enter a string: hi  
Enter a string: how are you?  
Enter a string: done  
The concatenation is: hi how are you?
```

**2. Sample Run**

```
Enter a string: where  
Enter a string: are  
Enter a string: you  
Enter a string: going?  
Enter a string: done  
The concatenation is: where are you going?
```

## Extra (Optional) Challenges

These are not required for credit but will help you practice and strengthen your problem-solving skills.

### Challenge 1: mTable (n)

Write a function **mTable** that takes one integer **n** and prints the multiplication table of **n**.

- **n** must be between 1 and 10.
- Print results from **n × 1** up to **n × 10**.

#### Sample Function Call:

```
n = 6
```

#### Expected Output

```
6 x 1 = 6
6 x 2 = 12
6 x 3 = 18
6 x 4 = 24
6 x 5 = 30
6 x 6 = 36
6 x 7 = 42
6 x 8 = 48
6 x 9 = 54
6 x 10 = 60
```

### Challenge 2: longestWord (sentence)

Write a function **longestWord** that takes a sentence and returns the longest word in it.

- If multiple words have the same longest length, return the first one that appears.
- Use loops and string methods.

#### Sample Function Call:

```
>>> longestWord("CS103 is really exciting")
"exciting"
>>> longestWord("I love Python")
"Python"
```

**Challenge 3: Integer to English Words (num)**

Convert a **non-negative integer** `num` to its English words representation.

Use a `while` loop to repeatedly process chunks of the number (thousands, millions, etc.).

**Function Signature:**

```
def numberToWords(num: int) -> str
```

**Sample Function Calls:**

Input: `num = 12345`

Output: "Twelve Thousand Three Hundred Forty Five"

**Challenge 4: guessTheNumber**

Write a function "`guessNumber`" that does not receive any input. The function will pick a random integer between **1** and **100** (both inclusive) and the user will try to guess the number in as few guesses as possible.

**Sample Input:**

The user will enter a guess until the correct guess is entered. The program will keep asking for a guess until the correct guess is entered. The program will also give **hints** to the user (number is too high, or too low).

**Sample Output:**

The output will be the number of guesses. Below is a sample run of how the program should proceed:

```
I'm thinking of a number between 1 and 100
Can you guess what the number is?
Enter your guess
50
Too low!
Enter your guess
75
Too low!
Enter your guess
87
Too low!
Enter your guess
93
You win!
You solved it in 4 guesses
```

---

To get attendance credit, finish Exercises 1–3.

If you finish early, try the optional challenges!