CS103 – Fall 2025
Lab03 Exercises

Objectives:

- To practice writing functions with multiple parameters and return values.
- To gain familiarity with conditionals and Boolean expressions.
- To apply mathematical formulas using Python's built-in math library.
- To reinforce problem-solving with real-world examples.

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

- Make a folder **Lab3** inside your **cs103fa25** folder.
- Create a new notebook inside your Lab3 folder (lab03.ipynb).

Required Exercises

Exercise 1: `bmiCalculator(wt, h)`

Write a function `bmiCalculator` that takes two floats, `wt` (weight in pounds) and `h` (height in inches), and returns the body mass index (BMI) as a float.
Formula:

$$BMI = \frac{703 \times wt}{h^2}$$

Sample Function Call:

```
>>> bmiCalculator(250, 72)

33.902391
```

Exercise 2: `nMultiplier(n)`

Write a function `nMultiplier` that takes an int `n` (0 < n < 10) and returns the value of `n + nn + nnn`.

Sample Function Call:

```
>>> nMultiplier(5)

615   # (5 + 55 + 555)
```

Exercise 3: simpleGrader(average)

Write a function `simpleGrader` that takes a float `average` and returns a Boolean (`True` or `False`).

- Return `True` if `average >= 70`
- Return `False` otherwise.

Sample Function Calls:

```
>>> simpleGrader(68)
False
```

Exercise 4: volCone(r, h)

Write a function `volCone(r, h)` that takes two nonnegative floats, the radius `r` of the cone's base and the height `h`, and returns the volume of the cone.

Formula:

$$V = \frac{1}{3}\pi r^2 h$$

Sample Function Calls:

```
>>> volCone(1, 1)
1.04719755
>>> volCone(2, 2)
8.3758040
>>> volCone(3, 2)
18.8495559
```

Extra (Optional) Challenges

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

Challenge 1: Write a function `quadraticSolver(a, b, c)` that returns the two solutions of the quadratic equation

$$ax^2 + bx + c = 0$$

- If the discriminant is negative, return `"No Real Roots"`.
- If the discriminant is zero, return one solution.
- Otherwise, return both solutions as floats.

Sample Function Calls:

```
>>> quadraticSolver(1, -3, 2)
(2.0, 1.0)
>>> quadraticSolver(1, 2, 1)
(-1.0,)
>>> quadraticSolver(1, 0, 4)
"No Real Roots"
```

Challenge 2: timeConverter(minutes)

Write a function `timeConverter(minutes)` that converts a given number of minutes into hours and minutes, returning a string formatted as `"H hours M minutes"`.

Sample Function Calls:

```
>>> timeConverter(130)
"2 hours 10 minutes"
>>> timeConverter(45)
"0 hours 45 minutes"
```

Challenge 3: twoSum(nums, target)

Write a function `twoSum(nums, target)` that takes a list of integers `nums` and an integer `target`.
Return a tuple of **two indices** whose numbers add up to the target.

- Assume there is exactly one solution.
- You cannot use the same element twice.

Sample Function Calls:

```
>>> twoSum([2, 7, 11, 15], 9)
(0, 1)        # because nums[0] + nums[1] = 2 + 7 = 9
>>> twoSum([3, 2, 4], 6)
(1, 2)
```

Challenge 4: taxCalculator(income)

Write a function `taxCalculator(income)` that takes a float `income` and returns the amount of tax owed based on these rules:

- If income ≤ 10,000 → tax = 0
- If 10,001 ≤ income ≤ 50,000 → tax = 10% of the amount over 10,000
- If 50,001 ≤ income ≤ 100,000 → tax = 4,000 (for first 50,000) + 20% of amount over 50,000
- If income > 100,000 → tax = 14,000 (for first 100,000) + 30% of amount over 100,000

Sample Function Calls:

```
>>> taxCalculator(8000)
0
>>> taxCalculator(20000)
1000.0
>>> taxCalculator(75000)
9000.0
>>> taxCalculator(120000)
20000.0
```

To get attendance credit, finish Exercises 1–4.
If you finish early, try the optional challenges!