=== Python Debugging Exercises === ## Exercise 1: average_calculator.py def calculate_average(grades): total = 0 for grade in grades: total += grade average = total / len(grades) return average student_grades = [85, 90, "A", 92, 88] avg = calculate_average(student_grades) print("Average grade:", avg) # Task: Debug the crash and make sure the function handles non-numeric values properly. ## Exercise 2: reverse_string.py def reverse_string(s): reversed = "" for i in range(0, len(s) + 1): reversed += s[i] return reversed print(reverse_string("hello")) # Task: Use the debugger to identify the IndexError and fix the loop. ## Exercise 3: find_max.py def find_max(numbers): max_value = 0 for num in numbers: if num > max_value: max_value = num return max_value print(find_max([-5, -10, -3])) # Task: Step through with the debugger and observe why this doesn't work with negative numbers. Fix the logic. ## Exercise 4: fibonacci.py def fibonacci(n): fib = [0, 1] for i in range(2, n): fib[i] = fib[i-1] + fib[i-2] return fib print(fibonacci(5)) # Task: Debug the crash caused by list index errors. Fix the logic using .append(). ## Exercise 5: append_item.py def append_item(item, item_list=[]): item_list.append(item) return item_list print(append_item(1)) print(append_item(2)) print(append_item(3)) # Task: Fix the unexpected behavior caused by the mutable default argument. # === Python Programming Exercises === ## Exercise 1: Palindrome Checker # Write a function `is_palindrome(s)` that checks whether a string is a palindrome. # Example: is_palindrome("radar") -> True ## Exercise 2: Prime Number Finder # Write a function `get_primes(n)` that returns a list of all prime numbers up to n. ## Exercise 3: Word Frequency Counter # Given a string of text, count the frequency of each word and return a dictionary. ## Exercise 4: Shopping Cart # Create a class `ShoppingCart` with methods to add, remove items, and calculate the total. ## Exercise 5: Student Gradebook # Build a program that stores students and their grades, and allows: # - Adding a new student # - Adding grades # - Getting average per student