# Module 7.6 Data Structures - Dictionary: Coding Questions

## Question 1: Write a function to count the frequency of words in a given string and return the result as a dictionary.

Hint: Split the string into words, then use a loop to count each word's occurrences. Store the counts in a dictionary.

## Question 2: Create a function that merges two dictionaries into one, where values from the second dictionary overwrite those from the first.

Hint: Use the update method or dictionary unpacking to merge the dictionaries.

## Question 3: Implement a function to invert a dictionary, swapping its keys and values.

Hint: Use a loop to iterate over the dictionary items and add them inverted to a new dictionary.

## Question 4: Write a function that takes a dictionary of employee names and their salaries and returns the name of the highest-paid employee.

Hint: Iterate over the dictionary to find the maximum salary and track the corresponding employee name.

## Question 5: Develop a function that takes a list of tuples (representing orders) and converts it into a dictionary where each key is an order number and its value is the order details.

Hint: Loop over the list of tuples, adding each as a key-value pair to a new dictionary.

## Question 6: Create a function to filter a dictionary by removing items with values below a specified threshold.

Hint: Use a dictionary comprehension or a loop with a conditional to create a filtered dictionary.

## Question 7: Implement a function that takes a dictionary of products (keys are product names, values are prices) and applies a discount to all prices.

Hint: Iterate over the dictionary, modifying each value to reflect the discount.

## Question 8: Write a function that categorizes a list of words by their first letter, returning a dictionary where each key is a first letter and its value is a list of words starting with that letter.

Hint: Use a loop to categorize words into lists, grouped by their starting letter, in a dictionary.

## Question 9: Develop a function that takes a dictionary of courses (keys are course names, values are lists of student names) and returns a dictionary where each student is a key and their value is a list of courses they are taking.

Hint: Invert the relationship in the original dictionary, aggregating courses by student name.

## Question 10: Create a function that accepts a dictionary representing a network graph (keys are node names, values are lists of connected nodes) and returns a list of all nodes reachable from a given starting node.

Hint: Use a recursive function or a loop with a stack/queue to traverse the graph and collect reachable nodes.