Table of Contents

## 1 FunctionDef fibonacci(n)

## 2 Function: fibonacci

The function of **fibonacci** is to compute the nth Fibonacci number using an iterative approach.

**parameters**: The parameters of this Function. · n: An integer representing the 0-indexed position of the Fibonacci number to compute.

**Code Description**: The function begins by checking if the input n is a negative number. If it is, a ValueError is raised, as the Fibonacci sequence is not defined for negative indices.

It then initializes two variables, a and b, to 0 and 1 respectively, which are the first two numbers in the Fibonacci sequence.

The function proceeds to loop n times. In each iteration, it simultaneously updates the values of a and b. The current value of b is assigned to a, and the sum of the old a and b is assigned to b. This process effectively steps through the Fibonacci sequence.

After the loop completes, the variable a holds the nth Fibonacci number, which is then returned as the result.

**Note**: The function requires a non-negative integer as input. The index n is 0-based, meaning fibonacci(0) returns the first number in the sequence (0), fibonacci(1) returns the second (1), and so on.

**Output Example**:

# Calling the function with n = 9  
fibonacci(9)  
  
# Expected return value  
34

## 3 FunctionDef invert\_dictionary(mapping)

## 4 Function: invert\_dictionary

The function of **invert\_dictionary** is to create a new dictionary by swapping the keys and values of a given dictionary, ensuring the original values are unique.

**parameters**: The parameters of this Function. · mapping: A dictionary mapping strings to integers. The docstring specifies that the values in this dictionary must be unique for the inversion to be successful.

**Code Description**: The function first checks if all values in the input mapping dictionary are unique. It does this by comparing the total number of values (len(mapping.values())) with the number of unique values, which is determined by converting the values into a set and getting its length (len(set(mapping.values()))). If the lengths are not equal, it means there are duplicate values, and the function raises a ValueError with the message “Values must be unique to invert dictionary”.

If all values are unique, the function proceeds to use a dictionary comprehension to construct a new dictionary. It iterates through each key-value pair of the input mapping and creates a new dictionary where the original value becomes the new key and the original key becomes the new value. This inverted dictionary is then returned.

**Note**: This function will raise a ValueError if the input dictionary contains duplicate values, as this would lead to key collisions in the resulting inverted dictionary.

**Output Example**:

# Given the input dictionary:  
input\_dict = {'alpha': 1, 'beta': 2, 'gamma': 3}  
  
# Calling invert\_dictionary(input\_dict) would return:  
{1: 'alpha', 2: 'beta', 3: 'gamma'}

## 5 FunctionDef is\_palindrome(text)

## 6 Function: is\_palindrome

The function of **is\_palindrome** is to determine if a given string is a palindrome, performing a case-insensitive comparison that ignores whitespace.

**parameters**: The parameters of this Function. · text: The string to check for palindrome properties.

**Code Description**: The function first normalizes the input text. It iterates through each character of the string, converts it to lowercase, and discards any character that is a whitespace character (like spaces, tabs, or newlines). These processed characters are then joined together to form a new string called normalized.

Finally, the function compares the normalized string with its reversed version, which is created using the slice notation [::-1]. It returns True if the normalized string is identical to its reverse, indicating it’s a palindrome, and False otherwise.

**Note**: The function’s palindrome check is case-insensitive and ignores all types of whitespace characters, not just simple spaces.

**Output Example**: Calling is\_palindrome("A man a plan a canal Panama") returns True.