Table of Contents

## 1 FunctionDef num(a, b)

**num**: The function of num is to return the sum of two numbers.

**parameters**: The parameters of this Function. · a: The first number to be added. · b: The second number to be added.

**Code Description**: The function num takes two arguments, a and b, and returns their sum using the + operator.

**Note**: The function assumes that the inputs a and b support the addition operation. If the inputs are not numbers, it may result in a TypeError.

**Output Example**: Calling num(5, 3) will return 8.

## 2 FunctionDef generate\_random\_integers(count, start, end)

**generate\_random\_integers**: The function of generate\_random\_integers is to return a list of pseudo-random integers within a specified range.

**parameters**: The parameters of this Function. · count: Number of integers to generate. · start: Inclusive lower bound for values. · end: Inclusive upper bound for values.

**Code Description**: The function generate\_random\_integers takes an integer count and optional start and end parameters, defaulting to 0 and 100 respectively. It first checks if count is negative, raising a ValueError if it is. Then, it checks if start is greater than end, and if so, swaps their values. Finally, it uses a list comprehension with random.randint(start, end) to generate a list of count random integers between start and end (inclusive) and returns the list.

**Note**: If start is greater than end, the function automatically swaps them to ensure the range is valid. The count parameter must be a non-negative integer.

**Output Example**: Calling generate\_random\_integers(5, 1, 10) might return [3, 8, 1, 5, 9].

## 3 FunctionDef choose\_random\_item(items)

**choose\_random\_item**: The function of choose\_random\_item is to select and return a random string from a given list of strings.

**parameters**: The parameters of this Function. · items: A list of strings to choose from.

**Code Description**: The function choose\_random\_item takes a list of strings named items as input. It first checks if the list is empty. If the list is empty, it raises a ValueError with the message “items must not be empty”. If the list is not empty, it uses the random.choice() function to select a random element from the list and returns that element.

**Note**: The function raises a ValueError if the input list is empty. It assumes that the random module is already imported.

**Output Example**: If items is ["apple", "banana", "cherry"], a possible return value is "banana".

## 4 FunctionDef shuffle\_copy(items)

**shuffle\_copy**: The function of shuffle\_copy is to return a shuffled copy of the input list without modifying the original list.

**parameters**: The parameters of this Function. · items: A list of integers.

**Code Description**: The function shuffle\_copy takes a list of integers items as input. It first creates a copy of the input list using list(items) and assigns it to the variable copy. Then, it uses the random.shuffle() function to shuffle the elements of the copy list in place. Finally, it returns the shuffled copy list.

**Note**: The function uses the random.shuffle method from the random module, which shuffles the list in place. Therefore, a copy of the original list is created to avoid modifying the original list.

**Output Example**: If the input is [1, 2, 3, 4, 5], a possible output is [3, 1, 5, 2, 4]. The order will vary due to the random shuffling.