

# Day 27: Testing

This problem is about unit testing.

Your company needs a function that meets the following requirements:

- For a given array of  $n$  integers, the function returns the index of the element with the minimum value in the array. If there is more than one element with the minimum value, it returns the smallest one.
- If an empty array is passed to the function, it raises an exception. A colleague has written this method. The implementation in Python is listed below. Implementations in other languages can be found in the code template.

```
def minimum_index(seq):
    if len(seq) == 0:
        raise ValueError("Cannot get the minimum value index from an empty sequence")
    min_idx = 0
    for i in range(1, len(seq)):
        if a[i] < a[min_idx]:
            min_idx = i
    return min_idx
```

A coworker has prepared functions that will perform the tests and validate return values. Finish the implementation of **3** classes to provide data and expected results for the tests.

Complete the following methods.

In the class `TestDataEmptyArray`:

- `get_array()` returns an empty array

In the class `TestDataUniqueValues`:

- `get_array()` returns an array of size at least 2 with all unique elements
- `get_expected_result()` returns the expected minimum value index for this array

In the class `TestDataExactlyTwoDifferentMinimums`:

- `get_array()` returns an array where the minimum value occurs at exactly 2 indices
- `get_expected_result()` returns the expected index

Take a look at the code template to see the exact implementation of functions that your colleague already implemented.

Note: The arrays are indexed from **0**.