**Exercises: Unit Testing**

Problems for exercises and homework for the ["C# OOP" course @ SoftUni"](https://softuni.bg/trainings/3585/csharp-oop-february-2022).

You can check your solutions here: <https://judge.softuni.org/Contests/1762/Unit-Testing-Exercises>

* **Database**

You are provided with a simple class – **Database**, which **stores integers**. **The constructor receives an array of numbers**, which is added into a private field. **The database** has the functionality to **add**, **remove** and **fetch all stored items**. Your task is to **test the class**. In other words, **write tests**, so you are sure its methods are working as intended.

**Constraints**

* Storing array's **capacity** must be **exactly 16 integers**
* If the size of the array is not 16 integers long, **InvalidOperationException** isthrown
* **The "Add()"** operation, should **add an element at the next free cell** (just like a stack)
* If there are 16 elements in the Database and try to add 17th, **InvalidOperationException** is thrown
* **The "Remove()"** operation, should support only removing an element **at the last index** (just like a stack)
* If you try to remove an element from an empty Database, **InvalidOperationException** is thrown
* **Constructors** should take integers only, and store them in the **array**
* **The "Fetch()" method** should return the elements as an **array**

For better understanding, check the provided skeleton.

**Hint**

Do not forget to **test the constructor(s)**. They are methods too!

* **Extended Database**

You already have a class - **Database**. We have **modified it** and added some **more functionality** to it. It supports **adding, removing, and finding People**. In other words, it **stores People**. There are two types of finding methods - first: "**FindById (long id)"** and the second one: "**FindByUsername (string username)"**. As you may guess, each person has **unique id** and **unique username**. Your task is to test **the provided project**.

**Constraints**

The database should have methods:

* **Add**
* If there are already users with this username, **InvalidOperationException** is thrown
* If there are already users with this id, **InvalidOperationException** is thrown
* **Remove**
* **FindByUsername**
* If no user is present by this username, **InvalidOperationException** is thrown.
* If the username parameter is null, **ArgumentNullException** is thrown
* Arguments are all **CaseSensitive**
* **FindById**
* If no user is present by this id, **InvalidOperationException** is thrown
* If negative ids are found, **ArgumentOutOfRangeException** is thrown

**Hint**

Do not forget to test the constructor(s). They are methods too! Also, keep in mind that all the functionality from the previous task still exists and you need to test it again!