# Testing

## Завдання - unit tests

### Prerequisites

З ***лабораторної роботи 1-2*** взяти приклад коду який ви обрали для покращення. Цей код має мати 20+ LOC (якщо менше - вам потрібно взяти ще якийсь код (метод, функцію, клас) додатково).

LOC - lines of code (кількість рядків коду)

### Опис задачі

1. Використовуючи код з лабораторної роботи 1-2 в якому ви виправляли design principles:
	1. Покрити ***оригінальний код*** unit-tests-ами. (має бути клас, метод, функція або набір класів, методів, функцій з 20+LOC).
	2. Запустити тести, навести скріншот виконання тестів і покриття. ***Покриття має бути 100%***
	3. Надати лістинг коду ваших тестів
	4. Покрити тестами змінений код. ***Покриття має бути 100%***
	5. Запустити тести, навести скріншоти покриття
	6. Надати лістинг коду тестів що тестують змінений код
	7. Будь-ласка наводьте код у виділених секціях для зручної навігації
2. **УВАГА! ВАЖЛИВО ДЛЯ УСПІШНОГО ВИКОНАННЯ** - Навести роздуми на ваш погляд - що змінилось в якості тестів, чи краще став код тестів, які зміни ви побачили.
3. Ваші тести мають запускатись з консолі консольною командою

## Як здати лабораторну роботу

Результатом виконання вашої лабораторної роботи має бути звіт (з титульною сторінкою, титульна - з рамкою, далі не вимагається). Звіт має бути структурований відповідно до завдань, містити код для перевірки.

Звіт надсилається на пошту викладача kkn\_vra@ztu.edu.ua у форматі PDF!

Subject має БУТИ ОБОВЯЗКОВО ‘{ваша група}, {СУЯІС - абревіатура предмета}, {ПІБ} - Лаб {N}’. - наприклад

ІСТ-20-1, СУЯІС, Вольський Ростислав - Лаб 6

## Appendix

### Корисні посилання:

* .NET - unit tests: <https://learn.microsoft.com/en-us/dotnet/core/testing/unit-testing-with-dotnet-test>
* JS - unit tests: <https://www.lambdatest.com/learning-hub/javascript-unit-testing>