

Homework. Do all the tasks in one VS solution.

NB! The goal of this homework is testing!

Exercise 1:

Create a new class called Exercises and add all the methods there. Then create a second class called ExercisesTesting and add all the tests there. Each exercise should have at least 4 tests, if needed then also more.

In that class create:

- 1) Method for joining together two strings. The strings are given as a parameter and result string is returned.
- 2) Method for generating and returning an array of 6 numbers where:
 - at least 2 numbers are even numbers, at least 2 numbers are odd numbers
 - 2 numbers are in a range from 0 to 20, 2 numbers from 30-60 and 2 numbers 100-200
 - one number divides by 5 (5,10,15,20,25 etc)

Use Random to generate this array.

Test all the requirements as well as you can. Examples:

<http://nunit.org/docs/2.6/collectionConstraints.html>

- 3) Method for calculating BMI. <https://www.bmi3d.com/formula.html>

This method takes height and weight as parameters and returns BMI value as text. Example:

CalculateBMI(165.3, 62) -> „Normal“

BMI	Weight Status
Below 18.5	Underweight
18.5 - 24.9	Normal
25.0 - 29.9	Overweight
30.0 and Above	Obese

Test all the 4 cases.

- 4) Create a method that takes string (a sentence) as input and puts * sign between words.
Test cases where there are multiple spaces between words; expected output is always one * sign.

Example:

*ReplaceSpaces(„I am a dog“) -> „I*am*a*dog“*

*ReplaceSpaces(„Nice Day“) -> „Nice*Day“*

Read: <https://www.dotnetperls.com/split>, <https://www.dotnetperls.com/string-join>

Exercise 2:

Create a new class called TimeCalculator

Create a method for calculating time zone differences which takes the amount of hours to add or subtract and calculates the new time. Input should be double.

Initial time value is 01.01.2000 00:00.

Result should contain date and time (hours and minutes, seconds are not needed).

Seconds are not needed. Method parameter has either + or – sign to determine whether to add or subtract values.

Examples:

FindTime(+2) -> 01.01.2000 02:00

FindTime(-1) -> 31.12.1999 23:00

FindTime(+25.5) -> 02.01.2000 01:30

Divide this task into submethods (addDay, subtractDay, findMinutesFromInput etc and test them separately).

Read: <https://www.dotnetperls.com/datetime>, <https://stackoverflow.com/questions/3577856/nunit-assert-areequal-datetime-tolerances>

Exercise 3

Create a class called EmployeeData.

We have a list of employee email addresses and we need to get their names based on the addresses.

The rules for finding the name:

- '.' is between first and last name
- '-' is between two first names
- '_' is between two last names.
- All names should start with a capital letter
- Last names should be written with '-' sign, first names separately.
- If name is less than 2 characters long, we replace it with „Unknown“

Example:

Email:	Name:
Mary.jones@contoso.com	Mary Jones
Tim-allen.Touring@hm.rt	Tim Allen Touring
Jim.jhon_son@g.eu	Jim Jhon-Son
t.ilves@eesti.eu	Unknown Ilves

The names should be kept in a separate list. Give the email values in the constructor (keep them in a list):

- andre.griffin@ou.eu
- jhon.snow@gt.eu
- tim-allen.toomingas@eu.eu
- hei-hoo.chee_choo@china.eu

- a.chin_chan@c.e

Create a method for finding the names from the addresses.

Read: <https://www.go4expert.com/articles/c-sharp-string-manipulation-tutorial-t30015/>

Tip:

There are many ways to do this.

- First goal is getting rid of the email ending. [Mary.jones@contoso.com](#) - > *Mary.jones*

One way for this is to determine the position of the @sign (*IndexOf(@)*) and then make a substring from beginning until the sign. Second option is to split the string where @ sign is.

- Separate first and family name part. *Mary.jones* -> *Mary jones*
- See if first and last name also contain special characters ('_' and '-')

Create submethods for different steps and test them separately too. Test all the 5 names to see if the methods work correctly. There should be at least 20 tests for this exercise. (Think of all the different things you can test).