Exercises about Lambda Expressions

- Solve them in Visual Studio.

Exercise 21.01: A Lambda List of Integers

	Solve the following exercises with Lambda Expressions. *As exercise, you use the Linq library with Lambda Expressions.*

- Create a list that contains integers.
- Show the list by looping thru it, and put a “;” in between. Make sure you don’t have one at the end.
- Filter that list, so you have only the odd numbers.
- Show the list by looping thru it, using the same method as before.
- Sort that result in descending order.
- Show the list by looping thru it, using the same method as before.
- Calculate the average of all the numbers in the resulting list.
- Show result.

Variant 1 (Exercise 22.01)

- Rewrite this, but you are using the fluent syntax notation of Linq.

Notes

Exercise 21.02: A Lambda List of Strings

	Solve the following exercises with Lambda Expressions. *As exercise, you use the Linq library with Lambda Expressions.*

- Create a list that contains texts.
- Show the list by looping thru it, with a “;” in between.
- Filter that list, so you have only the texts that contain the letter “e”, case insensitive.
- Show the list by looping thru it, using the same method as before.
- Sort that result in ascending order.
- Show the list by looping thru it, using the same method as before.
- Make the list in full capitals.
- Show the list by looping thru it, using the same method as before.

Variant 1 (Exercise 22.02)

- Rewrite this, but you are using the fluent syntax notation of Linq.

Notes

Exercise 21.03: A Lambda Dictionary of Integers

	Solve the following exercises with Lambda Expressions. *As exercise, you use the Linq library with Lambda Expressions.*

- Create a list that contains integers, there must be duplicates in it.
- Show the list by looping thru it, with a ";" in between.
- Filter that list, so you have only the even numbers.
- Show the list by looping thru it, using the same method as before.
- Create now a dictionary.
  - The key of the elements of the dictionary is the number itself.
  - The value of the elements it the number of times it occurs in the list of the even numbers.
- Sort this dictionary, on the frequency of the numbers. So you sort on the value of the dictionary. Highest number first. When there are equals, the dictionary must be sorted on the keys.
  - Think carefully on how you can do this.
- Show the result, every element of the dictionary on another line.
  - First the number (the key).
  - An arrow (-->).
  - Then the frequency (the value).

Variant 1 (Exercise 22.03)

- Rewrite this, but you are using the fluent syntax notation of Linq.

Notes

....

....

....

....

....

....

....