

Лабораторна №1: Вступ у функціональне програмування

Виконав: студент Германович Данило групи ІП-02

Завдання:

1. Створити програму для вирішення задач на улюбленій мові програмування
2. Встановити на власний комп'ютер середовище для запуску проектів на мові PureScript
3. Створити програму для вирішення задач мовою PureScript

Задачі:

- 1) Створити функцію Singleton, яка приймає на вхід один аргумент і повертає список із одного елемента (аргумента функції).
- 2) Створити функцію null, яка приймає на вхід список і повертає boolean з результатом перевірки чи пустий список.
- 3) Створити функцію spoc, яка приймає на вхід список і елемент, і повертає новий список із елементом, який доданий в кінець списку.
- 4) Створити функцію length, яка приймає на вхід список і повертає кількість елементів в списку.

Довідкова інформація :

як встановити PureScript:

[documentation/Getting-Started.md at master · purescript/documentation · GitHub](#)

Репозиторій - https://github.com/eluyema/fp_1

Код завдання №1:

```
// 1
function Singleton(param) {
    return [param];
}

const q = [ 1, 2, 3 ]
console.log(`1.1`, Singleton(q))

//2
function Null(list) {
    for(const element of list) {
        return false;
    }
    return true;
}

const emptyList = [];
const notEmptyList = [ "1", 2 ];
console.log('2.1', Null(emptyList));
console.log('2.2', Null(notEmptyList));

//3
function Snoc(list, el) {
    return [...list, el]
}

const list = [ 1, 2 ];
const el = "3"
console.log('3.1', Snoc(list, "3"))

//4
function Length(list) {
    let i = 0;
    for(const el in list) {
        i++;
    }
    return i;
}

const longList = [ 1, 2, "3", 4, "5", 6, 7, "8", 9 ]
console.log('4.1', Length(longList));
```

Скріншот виконання завдання №1:

```
PS D:\fp_lab_1> node .\favoriteLanguage.js
1.1 [ [ 1, 2, 3 ] ]
2.1 true
2.2 false
3.1 [ 1, 2, '3' ]
4.1 9
```

Скріншот виконання завдання №2 (результат команди spago run):

```
PS D:\fp_lab_1> npx spago run  
[info] Build succeeded.
```



Код завдання №3:

```
module Lab1 where
import Prelude

import Data.List (List(..), (:))

import Effect (Effect)
import Effect.Console (log)

singleton :: forall a. a -> List a
singleton a = a:Nil

null :: forall a. List a -> Boolean
null Nil = true
null _ = false

snoc :: forall a. List a -> a -> List a
snoc Nil v = v : Nil
snoc (a : rest) v = a : (snoc rest v)

lenght :: forall a. List a -> Int
lenght Nil = 0
lenght (a:Nil) = 1
lenght (_:a) = lenght(a) +1

test :: Effect Unit
test = do
  log $ show $ singleton (1:(1:(1:Nil)))
  log $ show $ null (1:(1:(1:Nil)))
  log $ show $ null Nil
  log $ show $ snoc (1:(1:(1:Nil))) 1
  log $ show $ snoc Nil 1
  log $ show $ lenght (1:(1:(1:Nil)))
  log $ show $ lenght (1:Nil)
  log $ show $ lenght Nil
```

Скріншот виконання завдання №3:

```
[info] Build succeeded.  
((1 : 1 : 1 : Nil) : Nil)  
false  
true  
(1 : 1 : 1 : 1 : Nil)  
(1 : Nil)  
3  
1  
0
```