KSFUPRO1KU, Functional Programming The IT University, Spring 2020

Exercise 1

Last update January 26, 2020

This exercise sheet must be handed in via LearnIt.

You are encouraged to solve this assignment individually.

Your name must be part of the filename, e.g., `FP-01-<name>.fsx`. An example: `FP-01-MadsAndersen.fsx`.

You can only upload one file and it must be of type `fs` or `fsx`.

It is important that you annotate your own code with comments. It is also important that you apply a functional style, i.e., no loops and no mutable variables.

Exercise 1.1 Write a function `sqr:int->int` so that `sqr x` returns x^2.

Exercise 1.2 Write a function `pow : float -> float -> float` so that `pow x n` returns x^n.

You can use the library function: `System.Math.Pow`.

Exercise 1.3 Solve HR, exercise 1.1

Exercise 1.4 Solve HR, exercise 1.2

Exercise 1.5 Solve HR, exercise 1.4

Exercise 1.6 Solve HR, exercise 1.5

Exercise 1.7 Solve HR, exercise 1.6

Exercise 1.8 Solve HR, exercise 1.7

Exercise 1.9 Solve HR, exercise 1.8

Exercise 1.10 Write a function `dup:string->string` that concatenates a string with itself.

You can either use + or ^. For example:

```
val dup : string -> string

> dup "Hi ";;
val it : string = "Hi Hi "
```

Exercise 1.11 Write a function `dupn:string->int->string` so that `dupn s n` creates the concatenation of `n` copies of `s`. For example:

```
val dupn : string -> int -> string

> dupn "Hi " 3;;
val it : string = "Hi Hi Hi "
```