# IN628 2019 - Introduction to Functional Programming in Python

The purpose of this practical is to give you experience with Python's higher-order functions, and encourage you to "think in Python". Try to write your solutions in the most succinct, functional, idiomatic way possible. Use map, filter and reduce, as appropriate. Use lambda expressions. Most critically: **no loops.** Don't rush through this practical; spend some time pondering it.

1. Assume that a passing grade on an exam is 50. Write a single Python statement that accepts a **list** of exam scores and returns all the values that did not pass. Your code should be something like this:

examScores = [65, 72, 41, 99, 32, 84]  
didNotPass = <***write your Python expression here...****>*  
print(list(didNotPass))

which would output [41, 32].

1. Use a single **reduce** expression to accept a list of strings and return the items of the list concatenated into a single string **in the reverse order.** Your code should be something like this:

**from** functools **import** reduce  
  
items = [**"ab"**, **"cd"**, **"ef"**, **"gh"**]  
reversed = reduce(***write the arguments to reduce here...***)  
print(reversed)

which would output ghefcdab.

1. Assume this string variable has been created:

starWars = "A long time ago in a galaxy far far away"

Write a single python statement that returns all the words in the sentence which are four or more characters in length, in upper case. You can accomplish this all in a single exporession (no loops!). Your code should look like:

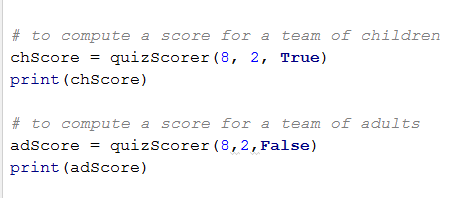
starWars = **"A long time ago in a galaxy far far away"**

longWords = <***write your Python expression here...>***  
print(list(longWords))

which will display ['LONG', 'TIME', 'GALAXY', 'AWAY'].

**Hint**: There is a built-in Python method which, when applied to a sentence, returns a **sequence** comprised of the words in the sentence (i.e. it splits on the space character and returns a sequence).

1. You are writing an application to compute scores for a Quiz Night. You want a flexible scoring algorithm to allow fair competition between teams composed of adults and teams composed of children. For Adult teams, the score is defined as 10 points for each correct answer minus 1 point for each incorrect answer. Child teams get 15 points per correct answer and lose 1/2 point for each incorrect answer. Build your scoring application using idiomatic Python. Your main logic should be a single function that will compute either the child team score or the adult team score, based on an input flag. Here is a fragment from my solution:



which prints:

