

## proj05: Functions and Lists

### Part I:

Create a function `divisors(num)` that returns the divisors of an integer.  
Call the function in another function `prime(num)` that returns true if the number is prime.

### Part II:

Create a function that takes two lists `intersection(lst1, lst2)`, say for example these two:

```
a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]
b = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]
```

and return a list that contains only the elements that are common between the lists (without duplicates). Make sure your program works on two lists of different sizes.

Extensions:

1. Randomly generate two lists to test this
2. Write this in one line of Python

### Part III:

Create a function `is_right` that determines whether or not 3 sides can make a right triangle by calling the functions above it:

<code>find_ab(side1, side2, side3)</code>	returns a list of the two shorter sides
<code>find_c(side1, side2, side3)</code>	returns the longest of the three sides
<code>square(side)</code>	returns the square of a number.
<code>pythagorean(a, b, c)</code>	returns true if $a^2 + b^2 = c^2$
<code>is_right(side1, side2, side3)</code>	returns true if the sides make up a right triangle