PPL- Part 1

1.

• Syntax: the syntax of a language can be expressed through the natural, brief and readable expression of a code given. the programmer controls the syntax and can extend it.

• Testing: how easy it is to test and verify that code is correct.

• Code Organization: the hierarchy of units (expressions, functions, modules, packages) and the way it's organized.

2.

a.

The function receives 2 variables X and Y and return their sum "X+Y".

This arithmetic expression can be done only with variable of type "Number".

And the signature of the function should look:

(x: Number, y: Number) => Number { return x+y }

b.

There are 2 options to the variables type:

1. The function receives String and return the first char in the String. The signature of the function will be: (x : String) => String { return x[0] }
2. The function recevies an array, the array type is T (because it can be any value, except undefined) and return the first value of the array.
   Which is the type T, the function will be: (x : T[]) => T { return x[0] }

c.

The function recevies X,Y we can see that X is boolean because if x is true the function returns y, else it return -y.

We can conclude that y is a Number because it's value can be Negative and positive which is only possible for type Number.

(x: Boolean , y: Number) => Number { x ? y : -y }

3.

"Shortcut Semantics" is a concept which we use to save running time and to make the code more efficient.

For example, the methods some and every are shortcut semantics.

Some stops and returns true when an element that satisfies the predicate is found. Every stops and returns false when an element that does not satisfy the predicate is found.