```
public static class
    private
    private int
    public
        /* initialization */

    public
        this

    public int         int
        return
```

```
add :: Integer, Integer -> Integer
add x y = x + y


cube :: Integer -> Integer
cube x = x * x * x


concat :: String, String -> String
concat x [] = x
concat x [y:ys] = concat [x:y] ys
```

A Difference

```
public static class NameEntry {
    private String name;
    private int[] ranks;
    public NameEntry() {
        /* initialization */
    }
    public setName(String name) {
        this.name = name;
    }
    public int getRank(int decade) {
        return ranks[decade];
    }
}
```

Pseudocode

```
add :: Integer, Integer -> Integer
add x y = x + y

cube :: Integer -> Integer
cube x = x * x * x

concat :: String, String -> String
concat x [] = x
concat x [y:ys] = concat [x:y] ys
```

Why Functional Programming?