

Laboratorium 5 – Swift

Programowanie mobilne

Zadanie 2. Deklarowanie zmiennych i interpolacja ciągu tekstowego

```
37
38 import Foundation
39
40 //z2
41 var points = 5.0;
42 print("Student has \(points) points");
43
Student has 5.0 points
```

Zadanie 3. Wyrażenie warunkowe.

```
44 //z3
45 var points = 5.0;
46 var satisfactionLevel = round(pow(Double.random(in:0...1), (110 - points) / 100) * 20)
47 var auraColor:String
48 switch satisfactionLevel {
49 case 0...5:
50 |   auraColor = "red"
51 case 6...10:
52 |   auraColor = "orange"
53 case 11...15:
54 |   auraColor = "purple"
55 case 16...20:
56 |   auraColor = "green"
57 default:
58 |   auraColor = ""
59 }
60
61 print(auraColor)
62
63
red
```

Zadanie 4 Klasy.

```
26 //z4
27 class Student{
28     var firstName:String
29     var lastName:String
30     var points:Double
31     init(firstName:String,lastName:String,points:Double){
32         self.firstName = firstName
33         self.lastName = lastName
34         self.points = points
35     }
36 }
```

Zadanie 5. Enkapsulacja i metody klasowe.

```
1 import Foundation
2
3 //z5
4 class Student{
5     private var name:String
6     private var lastName:String
7     private var points:Double
8     init(name:String,lastName:String,points:Double){
9         self.name = name
10        self.lastName = lastName
11        self.points = points
12    }
13    func printS(){
14        print("\(name) - \(lastName) - \(points)")
15    }
16
17    internal func showStudent(){
18        var satisfactionLevel = round(pow(Double.random(in:0...1), (110 - points) / 100) * 20)
19        var auraColor:String
20        switch satisfactionLevel {
21            case 0...5:
22                auraColor = "red"
23            case 6...10:
24                auraColor = "orange"
25            case 11...15:
26                auraColor = "purple"
27            case 16...20:
28                auraColor = "green"
29            default:
30                auraColor = ""
31        }
32        print("\(name) has a \(auraColor) face color")
33    }
34 }
35
36 let student = Student(name:"Mateusz",lastName:"Klos",points:2)
37 student.showStudent()
38
39
```

Mateusz has a purple face color

Zadanie 6. Dziedziczenie.

```
1  import Foundation
2
3  //z6
4  class Person{
5      var name:String
6      var lastName:String
7      init(name:String,lastName:String){
8          self.name = name
9          self.lastName = lastName
10     }
11 }
12
13 class Student:Person{
14     var points:Double
15     init(name:String,lastName:String,points:Double){
16         self.points = points
17         super.init(name: name, lastName: lastName)
18     }
19     func printS(){
20         print("\(name) - \(lastName) - \(points)")
21     }
22
23     internal func showStudent(){
24         var satisfactionLevel = round(pow(Double.random(in:0...1), (110 - points) / 100) * 20)
25         var auraColor:String
26         switch satisfactionLevel {
27             case 0...5:
28                 auraColor = "red"
29             case 6...10:
30                 auraColor = "orange"
31             case 11...15:
32                 auraColor = "purple"
33             case 16...20:
34                 auraColor = "green"
35             default:
36                 auraColor = ""
37         }
38         print("\(name) has a \(auraColor) face color")
39     }
40 }
41 let student = Student(name:"Mateusz",lastName:"Klos",points:2)
42 student.showStudent()
```

Mateusz has a red face color