Computer Science 316

iOS App Development

Lab Assignment #2

Due: Sunday, January 26, 2025 (Before midnight)

Given 4 employees with the following properties:

` name : Peter Long
id : 112-22-3011
jobtitle : Volunteer Worker
yearsOfService : 5
annualsalary: 1200.00

name : Martine Short
id : 116-23-6418
jobtitle : Manager
yearsOfService : 8
annualsalary: 78600.00

name : Susan Johnson
id : 123-32-3515
jobtitle : Receptionist
yearsOfService : 10
annualsalary: 38600.00

name : Paul Simon
id : 133-53-4019
jobtitle : System Support Analyst
yearsOfService : 8
annualsalary: 65000.00

Part I: tuple

Use Xcode to create a Playground file (and name it *Lab2_Tuple*) with the following specifications:

1. Create a new type named **Employee** using **typealias** and tuple data type as follow:

```
typealias Employee = (name: String, id: String, jobtitle: String, yearOfService: Int, annualsalary: Double)
```

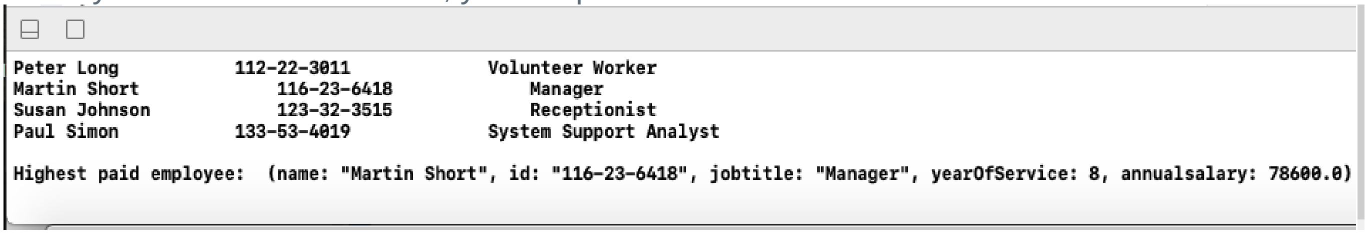
- Use the function call like this below to create a tuple object representing an employee:

```
Employee(name:"Peter Long", id: "112-22-3011", jobtitle: "Volunteer Worker",
                  yearOfService: 5, annualsalary: 1200.00)
```

- Create four tuples representing the four given employees above, and place them all into an array variable named **employees**.
- Use a for-loop to print out each employee's name , id and job title.
- Write a function which takes an array of Employee as a parameter (emps), and returns an Employee tuple object with the highest annual salary.
- Use the print statement and function call like this below for testing your highestSalary(emps) function:

  ```
  print ("Highest paid employee: ", highestSalary(emps: employees))
  ```

  When your codes are executed, your output results should look like this:



```
Peter Long           112-22-3011             Volunteer Worker
Martin Short             116-23-6418             Manager
Susan Johnson            123-32-3515             Receptionist
Paul Simon           133-53-4019             System Support Analyst

Highest paid employee:  (name: "Martin Short", id: "116-23-6418", jobtitle: "Manager", yearOfService: 8, annualsalary: 78600.0)
```

2. Leave the **Employee** type and **employees** array variable unchanged, now create the Dictionary variable to contain the same Employee objects, use a for-loop to copy each Employee object from the **employees** array into your Dictionary variable (name it **dict_employees**) using each employee's id as the key to associate with the employee object itself as the value**.** Repeat all the same tasks above with the dictionary variable: **dict_employees**.

   When you finish, close your Lab2_Tuple Playground file.

Part II: struct

Use Xcode to create the second Playground file (and name it *Lab2_Struct*). Declare a new data type named **Employee** using **struct**:

```
struct Employee {
          let name: String?
          let id: String?
          let jobtitle: String?
         let yearOfService: Int?
         let annualsalary: Double?

          init (name: String? = nil, id: String? = nil, title: String? = nil, year: Int? = nil,
```

```
        salary: Double? = nil){
            self.name = name
            self.id = id
            jobtitle = title
             yearOfService = year
            annualsalary = salary
        }//init
    } //struct
```

1. Repeat all the same tasks with an Array of Employee variable named **employees** as in PART I. All optional fields inside each employee object need to be unwrapped and printed properly in the outputs.

2. Repeat all the same tasks with a Dictionary of Employee variable named **dict_employees** as in PART I. All optional fields inside each employee object need to be unwrapped and printed properly in the outputs.

Submitting your work:

When you’re done, please copy your two Playground files into **Lab2** folder. Then compress **Lab2** folder and submit it via Lab 2 link on our Moodle course page by *Sunday, January 26, 2025 (before midnight).*