

Introduction to Swift

Exercise 2

Abhishek Maurya

Create a employee personal information structure and employee professional structure

The properties for personal :

```
employeeID
name
country(america,india,britain,japan,china)
address
hobbies(optional)
```

Properties for professional

```
employeeID
name
department(iOS, android, jvm, full stack, web)
branch(america,india,britain,japan,china)
experienced
```

TASKS:

1. create a third employee structure that contains the information from both based on common id.

```
• struct Employee {
    var employeeID: Int
    var name: String
    var department: String
    var branch: String
    var experience: Int
    var address: String
    var country: String
    var hobbies: String?
}

//Adding entries in Employee structure
var empArray: [Employee] = []

for personalEmp in personalArray {
    for professionalEmp in professionalArray {
        if (personalEmp.employeeID == professionalEmp.employeeID) {
            empArray.append(Employee(employeeID:
personalEmp.employeeID, name: personalEmp.name, department:
professionalEmp.department, branch: professionalEmp.branch,
experience: professionalEmp.experience, address: personalEmp.address,
country: personalEmp.country, hobbies: personalEmp.hobbies))
        }
    }
}
```

```
    }
  }

  print("Answer 1:")
  for index in empArray {
    print(index)
  }
}
```

2. write a function that takes the two structure and give me list of all the employee that live in certain country

- ```
func answer2(country: String) -> String{
 print("\nAnswer 2:")
 for personalEmp in personalArray {
 for professionalEmp in professionalArray{
 if (personalEmp.employeeID == professionalEmp.employeeID
 && personalEmp.country==country)
 {
 print(personalEmp)
 }
 }
 }
 return country
}
```

answer2(country: "India")

**3. write a function that give me list of all the employee that live in certain department**

- ```
func answer3(department: String) -> String{
  print("\nAnswer 3:")
  for emp in empArray {
    if (emp.department==department)
    {
      print(emp)
    }
  }
  return department
}
```

answer3(department: "iOS")

4. write a function that give me list of all the employee that live in same country and work in the same branch.

- ```
func answer4(ans: String) -> String{
 print(ans)
 for emp1 in empArray {
 for emp2 in empArray {
 if (emp1.country == emp2.branch) {
 print(emp1)
 }
 }
 }
}
```

```

 }
 return ans
}
answer4(ans: "\nAnswer 4:")

```

5. write a function that return me list of all the employee name that has a hobby and with their experience.

```

• func answer5(ans :String) -> String{
 print(ans)
 for emp in empArray {
 print("ID: \(emp.employeeID), Name: \(emp.name), Hobbies:
 \(emp.hobbies ?? "NA"), Experience: \(emp.experience)")
 }
 return ans
}
answer5(ans: "\nAnswer 5:")

```

6. write a function that return me list of all the employee name that starts with any "S"

```

• func answer6(prefix: String) -> String{
 print("\nAnswer6")
 for emp in empArray {
 if (emp.name.prefix(1)==prefix) {
 print(emp)
 }
 }
 return prefix
}
answer6(prefix: "S")

```