**Q: 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)**

**experience**

**TASKS:**

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

**enum** country{

**case** america,india,britain,japan,china,defaultCountry

}

**enum** department{

**case** ios,android,jvm,fullStack,web,defaultDepartment

}

**enum** branch{

**case** america,india,britain,japan,china,defaultBranch

}

**struct** EmployeePersonal{

**var** employeeId : Int

**var** name : String

**var** Country = country.defaultCountry

**var** address : String

**var** hobbies : String?

}

**struct** EmployeeProfessional{

**var** employeeId : Int

**var** name : String

**var** Department = department.defaultDepartment

**var** Branch = branch.defaultBranch

**var** experience : Int

}

**struct** Employee{

**var** employeeId : Int

**var** name : String

**var** Country = country.defaultCountry

**var** address : String

**var** hobbies : String?

**var** Department = department.defaultDepartment

**var** Branch = branch.defaultBranch

**var** experience : Int

}

**var** personalArray : [EmployeePersonal] = [EmployeePersonal(employeeId: 1, name: "Rajat", Country: .india, address: "XYZ", hobbies: "Coding"),

EmployeePersonal(employeeId: 3, name: "Anupam", Country: .japan, address: "ABC", hobbies: "Reading"),

EmployeePersonal(employeeId: 2, name: "Sreyansh", Country: .india, address: "QWE", hobbies: **nil**),

EmployeePersonal(employeeId: 4, name: "Subarno", Country: .britain, address: "VCX", hobbies: "Cricket"),

EmployeePersonal(employeeId: 8, name: "Nilesh", Country: .america, address: "LKJH", hobbies: "Tennis"),]

**var** professionalArray : [EmployeeProfessional] = [EmployeeProfessional(employeeId: 2, name: "Sreyansh", Department: .ios, Branch: .india, experience:4),

EmployeeProfessional(employeeId: 4, name: "Subarno", Department: .android, Branch: .britain, experience:5),

EmployeeProfessional(employeeId: 1, name: "Rajat", Department: .ios, Branch: .india, experience:5),

EmployeeProfessional(employeeId: 3, name: "Anupam", Department: .jvm, Branch: .japan, experience:2),

EmployeeProfessional(employeeId: 8, name: "Nilesh", Department: .fullStack, Branch: .america, experience:4),]

**var** thirdArray : [Employee] = []

**for** personalItem **in** personalArray{

**for** profesionalItem **in** professionalArray{

**if** personalItem.employeeId == profesionalItem.employeeId{

**var** tempEmployee : Employee = Employee(employeeId: personalItem.employeeId, name: personalItem.name,Country: personalItem.Country, address: personalItem.address, hobbies: personalItem.hobbies, Department: profesionalItem.Department,Branch: profesionalItem.Branch, experience: profesionalItem.experience)

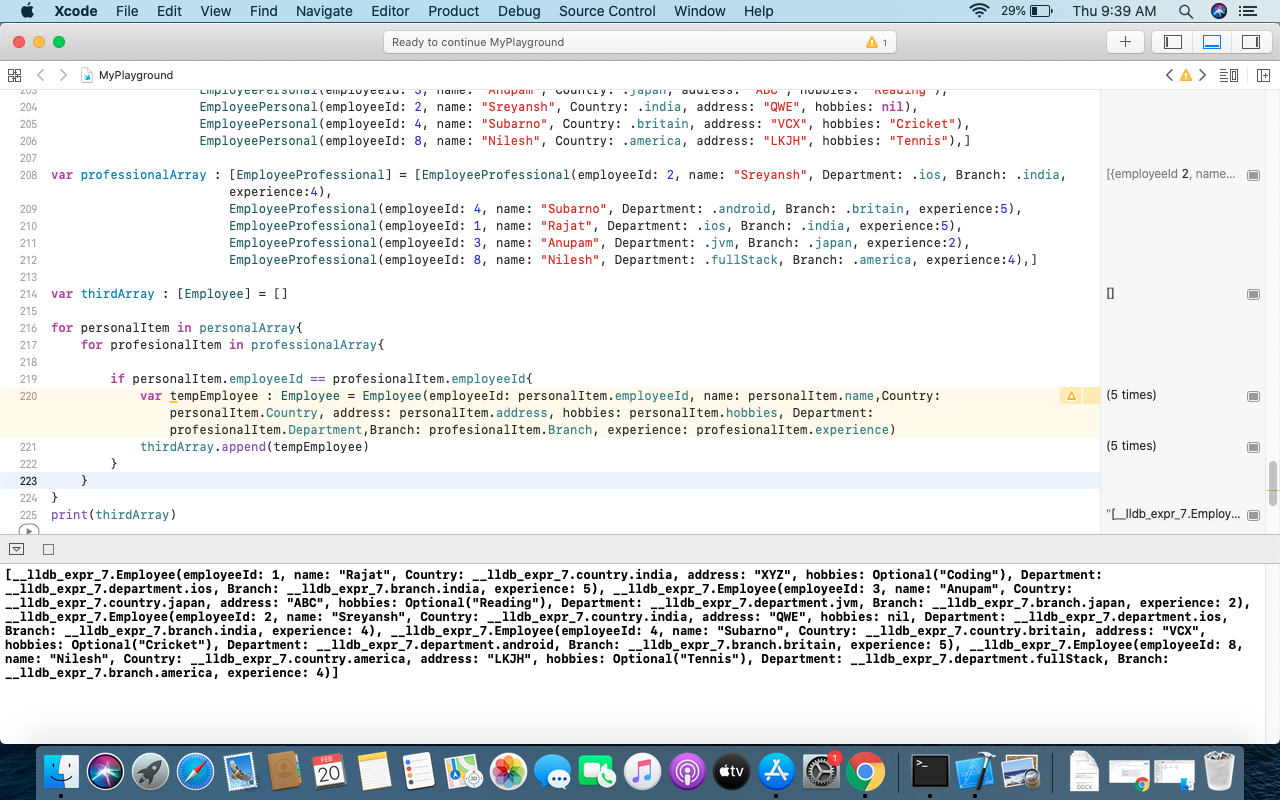
thirdArray.append(tempEmployee)

}

}

}

print(thirdArray)



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

**func employeeCountree(\_ empArr : [Employee], \_ country : country) {**

**for item in empArr{**

**if item.Country == country{**

**print(item)**

**}**

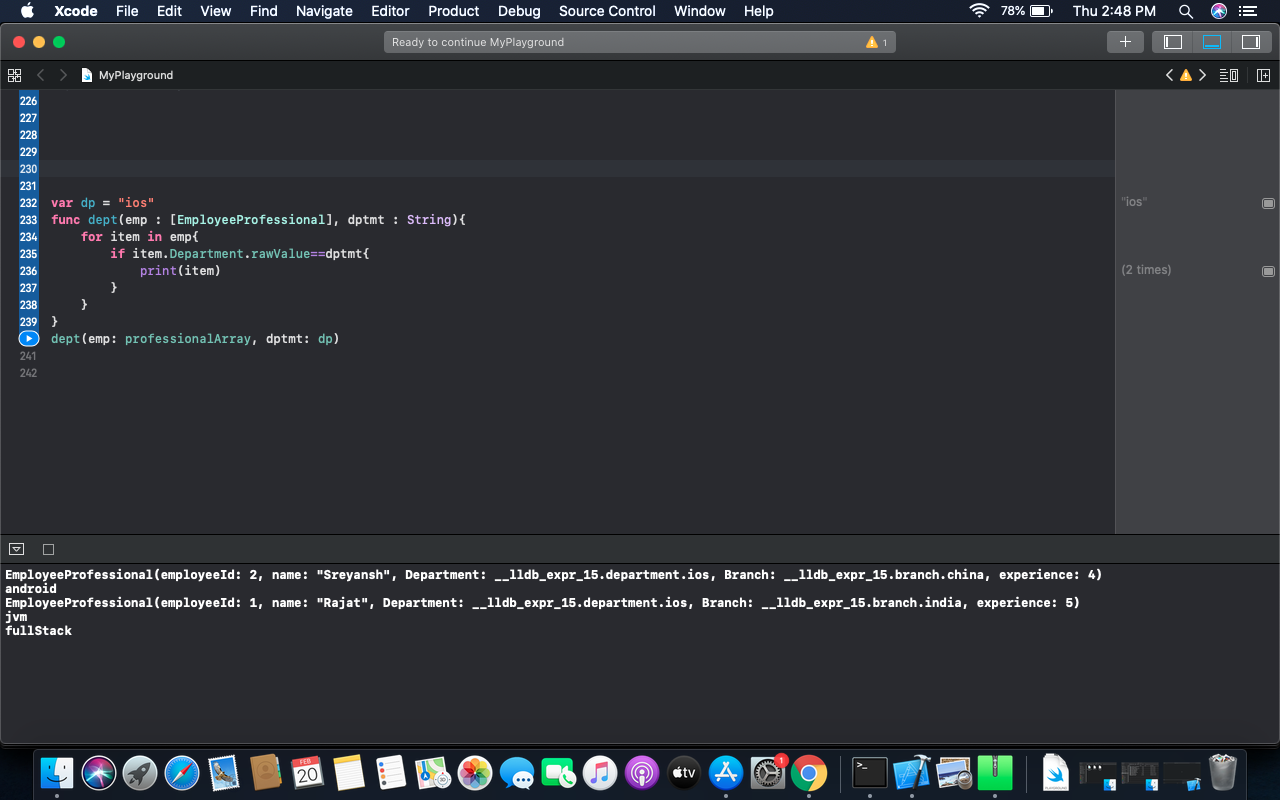
**}**

**}**

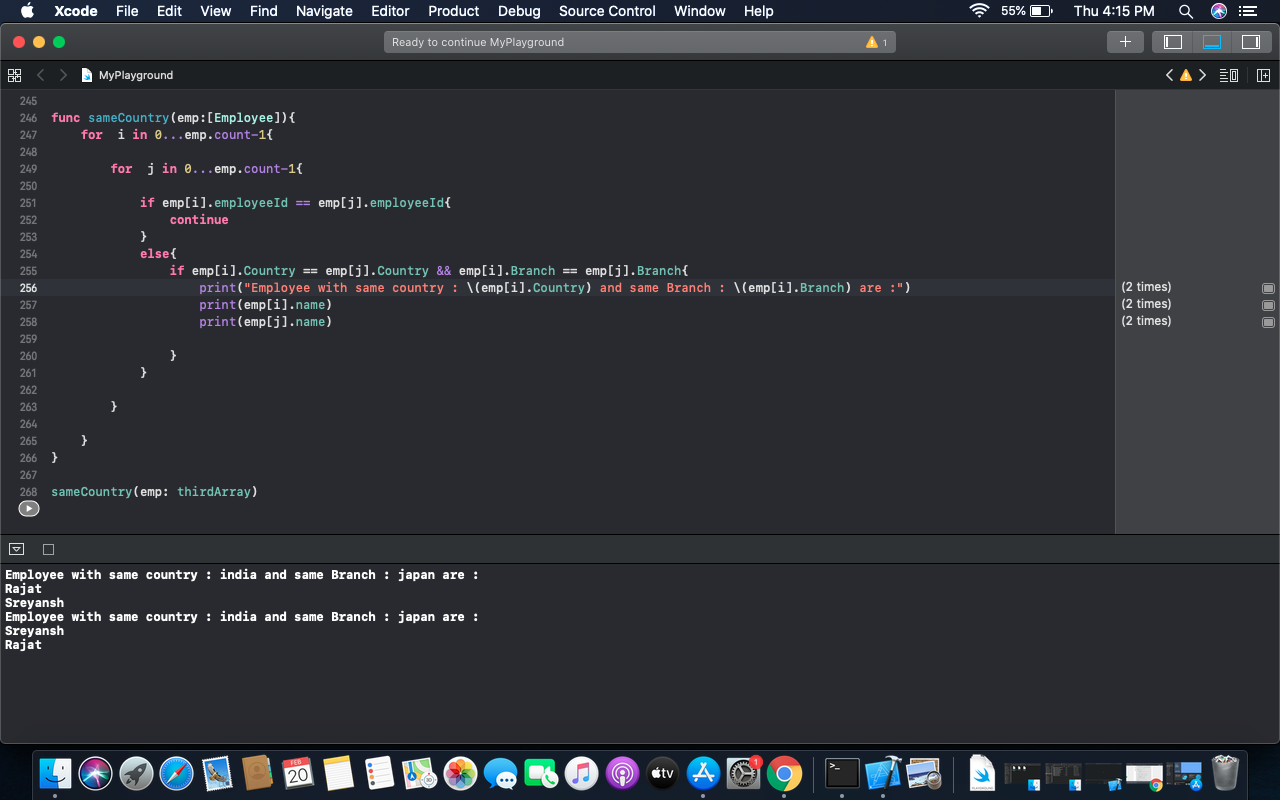
**let ecountry = country.india**

**employeeCountree(thirdArray, ecountry)**

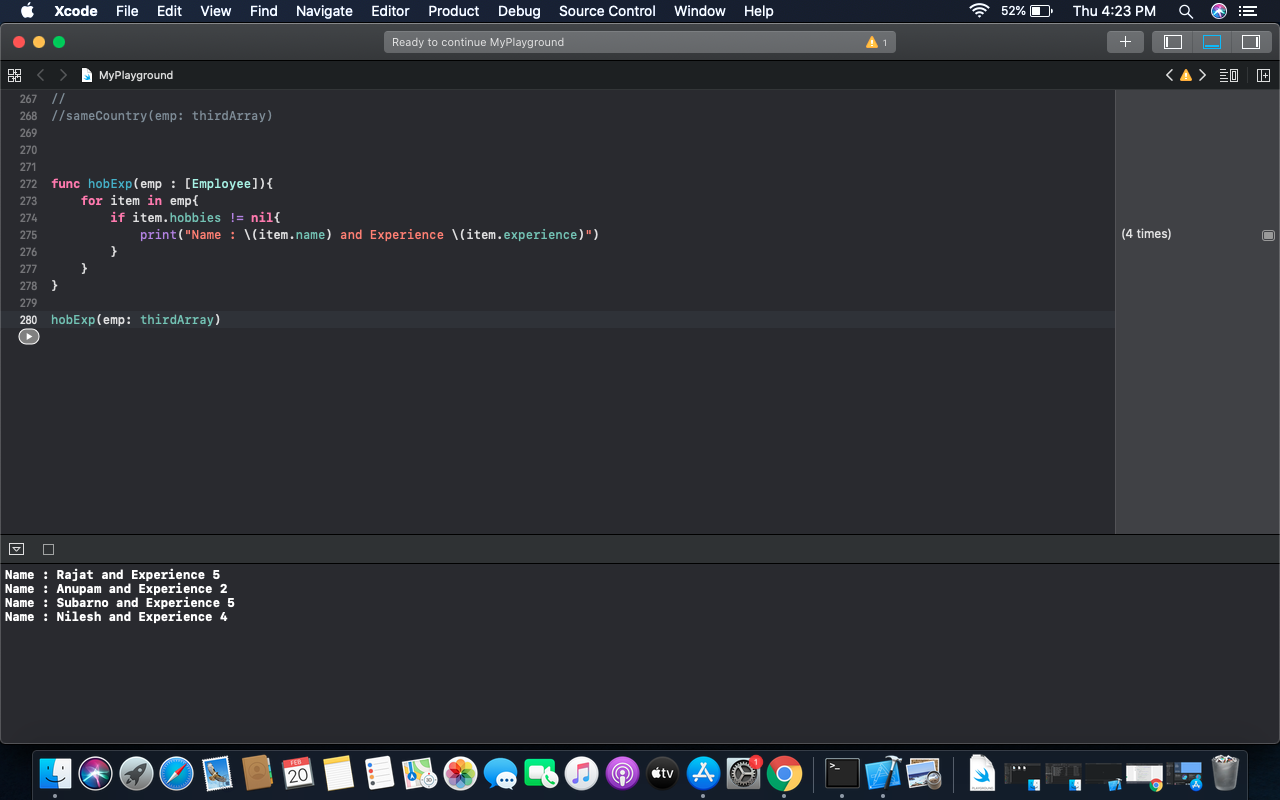
**3. write a function that give me list of all the employee that live in certain department**

****

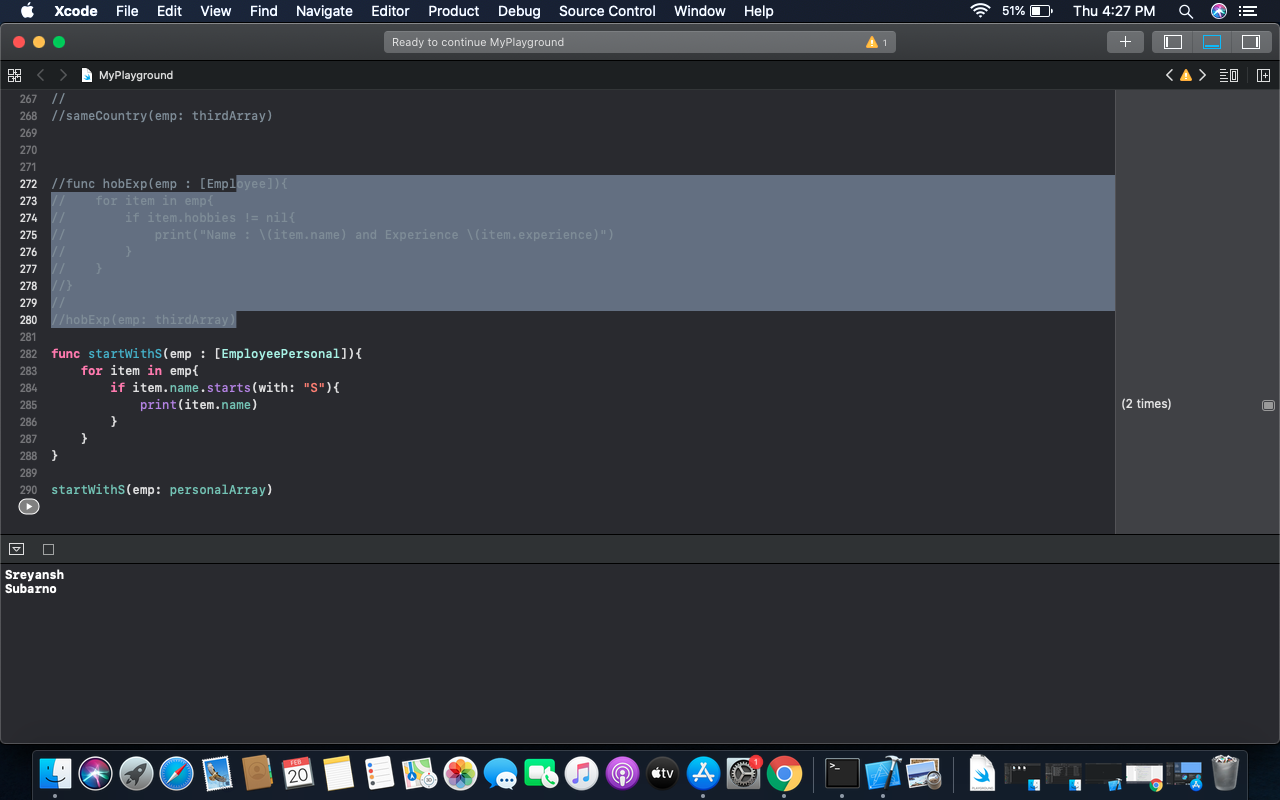
**4. write a function that give me list of all the employee that live in same country and work in the same branch.**

****

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

****

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

****