RUBY PROGRAMMING LAB:9

NAME: S KAMALESH KUMAR

REG NO: 18MIS1080

1. Define a module **STUDENT** with the following details:

Variables : **reg number, name**

Functions : **read()-** to initialize the values

**Display()-** to display the values

Define a another module **FACULTY** with the following details:

Variables : **employee Id, name, school**

Functions : **read1()-** to initialize the values

**Display1()-** to display the values

Define a class **COURSE ALLOCATION** with the details **: course code, Slot** and includes all the details of both the modules **STUDENT** and **FACULTY**.

Perform the following operation :

1. Create N objects of COURSE ALLOCATION class. Initialize the values using Constructor .
2. Display all the records
3. Display the faculty names who are handling same subjects. For each subject display all faculty names.
4. How many students are registered for each faculty for a specific course. Accept the course code from the user.
5. List out all course name with the Faculty name in each slot.
6. Display total number of courses handled by the faculty of each school.

Code:

Stud.rb

module Student

attr :name,:regno

def Student.read(name,regno)

    @name = name

    @regno= regno

end

end

def Student.display()

    puts "Student name:#{@name}"

    puts "Student Registration number:#{@regno}"

end

end

faculty.rb

module Faculty

attr:name1,:empid,:school

def Faculty.read1(empid,fname,school)

    @empid=empid

    @fname=fname

    @school=school

end

def Faculty.display1()

    puts "Faculty name:#{@fname}"

    puts "Faculty id:#{@empid}"

    puts "Faculty school:#{@school}"

end

end

lab9.rb

$LOAD\_PATH << '.'

require "stud"

require "faculty"

class Course\_allocation

include Student

include Faculty

    def initialize(coursecode,slot,regno,name,empid,fname,school)

        @coursecode = coursecode

        @slot = slot

        @fname = fname

        @name = name

        Student.read(name,regno)

        Faculty.read1(empid,fname,school)

    end

    def display

        Student.display()

        Faculty.disp()

    end

    def slot

        if @slot.eql?("A1")

            puts "#{@coursecode} handling by #{@fname} in A1 slot"

        end

    end

end

print("Enter the number of course allocation:")

n=gets.to\_i

ar=[]

for i in 0...n do

print("Enter course code for the #{i+1} allocation:")

coursecode=gets().to\_s

print("Enter slot for the #{i+1} allocation:")

slot=gets().to\_s

print("Enter Registration number for the #{i+1} allocation:")

regno=gets().to\_i

print("Enter Student name for the #{i+1} allocation:")

name=gets().to\_s

print("Enter Faculty id for the #{i+1} allocation:")

empid=gets().to\_i

print("Enter faculty name for the #{i+1} allocation:")

fname=gets().to\_s

print("Enter faculty school name for the #{i+1} allocation:")

school=gets().to\_s

c = Course\_allocation.new(coursecode,slot,regno,name,empid,fname,school)

ar.push(c)

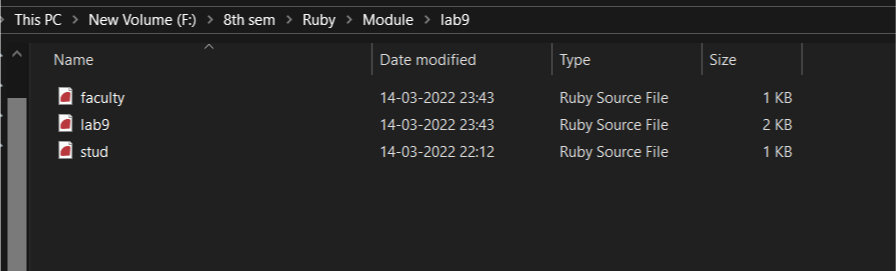
end

for i in 0...n

    ar[i].display

end

**file saving format:**



Output:

