**ViewController**

**import** UIKit

**import** CoreData

**class** ViewController: UIViewController, UIPickerViewDelegate,UIPickerViewDataSource {

**let** department = ["Manager","Sales","IT"]

**@IBOutlet** **weak** **var** dp: UIDatePicker!

**@IBOutlet** **weak** **var** pickerview: UIPickerView!

**@IBOutlet** **weak** **var** name: UITextField!

**@IBOutlet** **weak** **var** id: UITextField!

**override** **func** viewDidLoad() {

**super**.viewDidLoad()

        pickerview.delegate = **self**

        pickerview.dataSource = **self**

    }

**func** pickerView(\_ pickerView: UIPickerView, titleForRow row: Int, forComponent component: Int) -> String? {

**return** department[row]

    }

**func** pickerView(\_ pickerView: UIPickerView, didSelectRow row: Int, inComponent component: Int) {

    }

**func** numberOfComponents(in pickerView: UIPickerView) -> Int {

**return** 1

    }

**func** pickerView(\_ pickerView: UIPickerView, numberOfRowsInComponent component: Int) -> Int {

**return** department.count

    }

**@IBAction** **func** insert(\_ sender: **Any**) {

**let** deptindex = pickerview.selectedRow(inComponent: 0)

**let** dept = department[deptindex]

**let** context = (UIApplication.shared.delegate **as**! AppDelegate).persistentContainer.viewContext

**let** newitem = NSEntityDescription.insertNewObject(forEntityName: "Studenttb", into: context)

        newitem.setValue(Int(id.text!), forKey: "id")

        newitem.setValue(name.text, forKey: "name")

        newitem.setValue(dept, forKey: "dept")

        newitem.setValue(dp.date, forKey: "dob")

**do**{

**try**! context.save()

            print("Inserted!!")

**let** fetch = NSFetchRequest<Studenttb>(entityName: "Studenttb")

**var** data = [Studenttb]()

**do**{

                data = **try**! context.fetch(fetch)

**for** i **in** data{

                    print(i.id)

                    print(i.name!)

                    print(i.dept!)

                    print(i.dob!)

                    print("----------------")

                }

            }

        }

    }

}

**DBHelper**

**import** Foundation

**import** CoreData

**import** UIKit

**class** DBHelper{

**static** **var** instance = DBHelper()

**let** context = (UIApplication.shared.delegate **as**! AppDelegate).persistentContainer.viewContext

**func** getdata()->[Studenttb]{

**var** data = [Studenttb]()

**let** fetch = NSFetchRequest<Studenttb>(entityName: "Studenttb")

**do**{

            data = **try**! context.fetch(fetch)

        }

**return** data

    }

}

**ListViewController**

**import** UIKit

**class** ListViewController: UIViewController, UITableViewDelegate,UITableViewDataSource {

**var** data = [Studenttb]()

**@IBOutlet** **weak** **var** tableview: UITableView!

**override** **func** viewDidLoad() {

**super**.viewDidLoad()

        data = DBHelper.instance.getdata()

    }

**func** tableView(\_ tableView: UITableView, numberOfRowsInSection section: Int) -> Int {

**return** data.count

    }

**func** tableView(\_ tableView: UITableView, cellForRowAt indexPath: IndexPath) -> UITableViewCell {

**let** cell = tableView.dequeueReusableCell(withIdentifier: "TableViewCell",for: indexPath)**as**! TableViewCell

        cell.id.text = String(data[indexPath.row].id)

        cell.name.text = data[indexPath.row].name

        cell.dept.text = data[indexPath.row].dept

**let** dateformat = DateFormatter()

        dateformat.dateFormat = "dd/MM/yyyy"

**let** dob = data[indexPath.row].dob

**if** **let** dob = dob{

            cell.dob.text = dateformat.string(from: dob)

        }

**return** cell

    }

}

**TableViewCell**

**import** UIKit

**class** TableViewCell: UITableViewCell {

**@IBOutlet** **weak** **var** dob: UILabel!

**@IBOutlet** **weak** **var** dept: UILabel!

**@IBOutlet** **weak** **var** name: UILabel!

**@IBOutlet** **weak** **var** id: UILabel!

**override** **func** awakeFromNib() {

**super**.awakeFromNib()

        // Initialization code

    }

**override** **func** setSelected(\_ selected: Bool, animated: Bool) {

**super**.setSelected(selected, animated: animated)

        // Configure the view for the selected state

    }

}