

solutions

full source code at:

https://github.com/jonasfrey/sqlite_tutorial/tree/main/excerice_week8

JPAExample.java

```
...
    EntityManagerFactory factory =
        Persistence.createEntityManagerFactory("university");
    EntityManager em = factory.createEntityManager();

    TypedQuery<Instructor> q1 =
        em.createQuery("SELECT e FROM Instructor e",
Instructor.class);
    List<Instructor> instructorList = q1.getResultList();

    for (Instructor instructor : instructorList) {
        System.out.println(instructor);
    }

    TypedQuery<Department> q2 =
        em.createQuery("SELECT e FROM Department e",
Department.class);
    List<Department> departmentList = q2.getResultList();

    for (Department o_department : departmentList) {
        List<Instructor> a_o_instructor_filtered =
o_department.f_a_o_instructor();

        float n_sum = 0;

        for (Instructor o_instructor_filtered :
a_o_instructor_filtered) {
            n_sum += o_instructor_filtered.salary;
            System.out.println(o_instructor_filtered);
        }

        System.out.println("average salary:");
        System.out.println(n_sum/a_o_instructor_filtered.size());
    }

    // many to many connections
    for (Instructor instructor : instructorList) {
        System.out.println(instructor.f_a_o_department());
    }
}
```

```
        for (Department o_department : departmentList) {
            System.out.println(o_department.f_a_o_instructor());
        }

        em.close();
        factory.close();

        ...
    }
```

Instructor.java

```
public Integer id;
public String name;
public String dept_name;
public float salary;

public Instructor() {
    super();
}

public List<Department> f_a_o_department() {

    EntityManagerFactory factory =
        Persistence.createEntityManagerFactory("university");
    EntityManager em = factory.createEntityManager();

    TypedQuery<Department> q1 =
        em.createQuery("SELECT e FROM department e",
Department.class);
    List<Department> departmentList = q1.getResultList();

    List<Department> a_o_department_filtered = departmentList.stream()
        .filter(p -> p.dept_name.equals(this.dept_name)
    ).collect(Collectors.toList());

    return a_o_department_filtered;
}

@Override
public String toString() {
    return "Instructor [id=" + id +
        ", name=" + name +
        ", dept_name=" + dept_name +
        ", salary=" + salary + "];"
}
```

Department.java

```
...
    public String dept_name;
    public String building;
    public float budget;

    public Department() {
        super();
    }
    public List<Instructor> f_a_o_instructor() {

        EntityManagerFactory factory =
            Persistence.createEntityManagerFactory("university");
        EntityManager em = factory.createEntityManager();

        TypedQuery<Instructor> q1 =
            em.createQuery("SELECT e FROM Instructor e",
Instructor.class);
        List<Instructor> instructorList = q1.getResultList();

        List<Instructor> a_o_instructor_filtered = instructorList.stream()
            .filter(p -> p.dept_name.equals(this.dept_name)
).collect(Collectors.toList());

        return a_o_instructor_filtered;
    }

    @Override
    public String toString() {
        return "Department [" +
            " dept_name=" + dept_name +
            ", building=" + building +
            ", numeric=" + budget + " ]";
    }
...

```