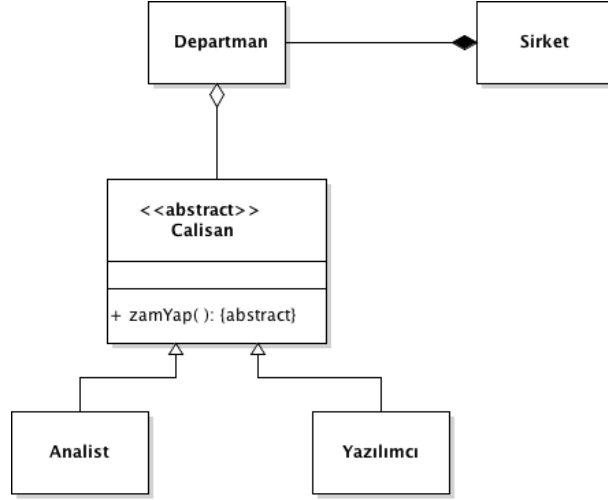


## NYP LAB-2



UML diyagramı genel görünümü verilen sistemde, **bir Şirketin 4 Departmanı, her Departmanın en fazla 3 Çalışanı** bulunmaktadır. Dizi yapısı kullanarak verileri tutmanız beklenmektedir.

**Calisan** sınıfından kalıtımla **Analist** ve **Yazılımcı** sınıfları oluşturulmaktadır. **Calisan** sınıfı, çalışanların ismini, kaç yıldır görev yaptığını ve maaş miktarını tutmaktadır. Alt sınıf olan **Analist** sınıfı da aynı bilgileri kullanırken, **Yazılımcı** sınıfı, bu bilgilere ek olarak çalışanın aldığı toplam sertifika sayısı bilgisini de tutmaktadır.

**kendiniTanıt** metodu ile çalışana ait bilgiler konsolda çıktısı görüldüğü şekilde ekrana yazdırılmalıdır. Yazılımcılar için ek olarak sertifika sayısı bilgisinin de yazdırıldığına dikkat ediniz.

**zamYap** metodu **abstract** olarak tanımlanmıştır. Bu sebepten **Calisan** sınıfı **abstract** olmaktadır. **zamYap** metodu, alt sınıflar olan **Analist** ve **Yazılımcı** sınıflarında implement edilmelidir. Yılda sadece 1 kez zam yapılması sebebiyle, **zamYap** metodunun çalışmasıyla görev süresinin 1 yıl artması gerekmektedir. Zam oranı **Analist** için %10, **Yazılımcı** için %15 olmalı, **zamYap** metodunun çalışmasıyla maaşlar güncellenmelidir.

**Test** sınıfında oluşturulan ve bir dizi içinde tutulan çalışanlar, **calisanEkle** metodu ile sırasıyla departmanlara yerleştirilecektir. **calisanEkle** metodu, ilk departmanın dolmasıyla sıradaki departmana ekleme yapacak şekilde tasarlanmalıdır.

**depMaasOrt** metodu ile numarası verilen departmanda çalışanların ortalama maaşları ekrana yazdırılmalıdır.

**maxGorevYili** metodu, şirkette en uzun süredir çalışan kişilerin isimlerini konsola yazdırmalıdır.

**sirketTanıt** metodu, her departmanda kaç adet Analist, kaç adet Yazılımcı bulunduğu bilgisini listelemelidir. Bu kısımda **instanceof** operatörünü kullanarak, departman çalışanlarının Analist veya Yazılımcı sınıflarına ait olma durumlarını belirleyebilirsiniz.

#### **instanceof kullanımı:**

**if**(nesneAdi instanceof sinifAdi) -> eğer nesne, adı verilen sınıfa ait ise **true** döner, aksi halde **false** döner.

**if**(calisanNesnesi instanceof Calisan) -> **true**

**if**(analistNesnesi instanceof Calisan) -> **true**

**if**(analistNesnesi instanceof Yazılımcı) -> **false**

#### **OUTPUTS:**

```
İsmim: Analist-4, 6 yıldır çalışıyorum. 6500.0 lira maaş alıyorum.
İsmim: Yazılımcı-6, 9 yıldır çalışıyorum. 7500.0 lira maaş alıyorum.7
adet sertifikam var.
*****
zamYap metodu çalışıyor...
*****
İsmim: Analist-4, 7 yıldır çalışıyorum. 6825.0 lira maaş alıyorum.
İsmim: Yazılımcı-6, 10 yıldır çalışıyorum. 8625.0 lira maaş
alıyorum.7 adet sertifikam var.
*****
Çalışanlar departmanlara yerleştiriliyor...
*****
3 numaralı departmanda çalışanların maaş ortalaması: 7725.0
*****
Görev yılı en fazla olan kişilerin isimleri:
Yazılımcı-6
Analist-5
*****
0 numaralı departmanda 2 tane Analist, 1 tane Yazılımcı vardır.
1 numaralı departmanda 0 tane Analist, 3 tane Yazılımcı vardır.
2 numaralı departmanda 2 tane Analist, 1 tane Yazılımcı vardır.
3 numaralı departmanda 1 tane Analist, 1 tane Yazılımcı vardır.
```