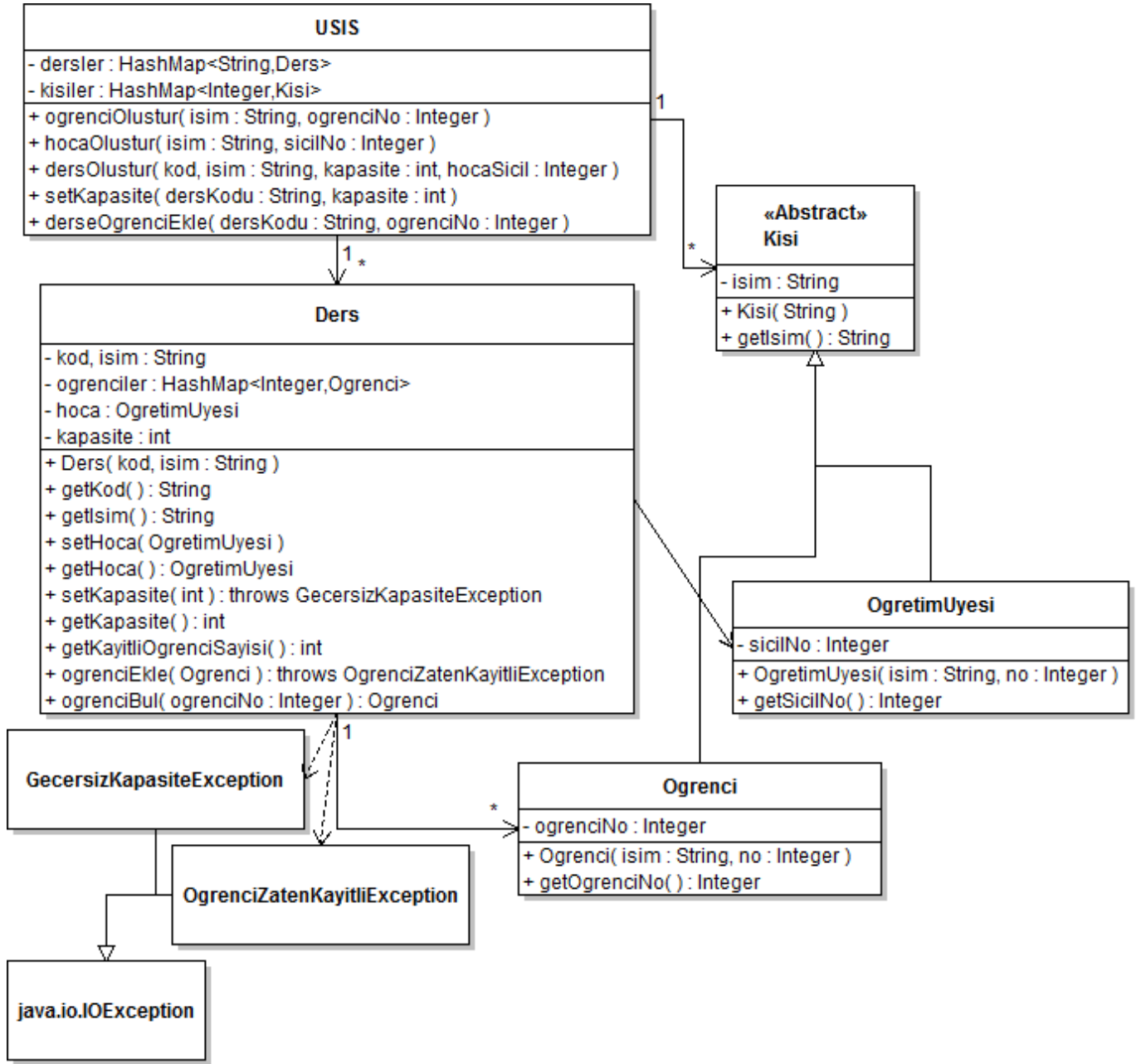


## 0112562 NESNEYE DAYALI PROGRAMLAMA Gr.1 2. ARA SINAVI ÇÖZÜMLERİ

Soruları yanıtlandırırken aşağıdaki UML sınıf şemasının kullanılması istenmiştir:



**Soru 1:** (20p) Kisi, OgretimUyesi ve Ogrenci sınıflarını kodlayınız.

```
package ara2;

public abstract class Kisi {
    private String isim;

    public Kisi(String isim) {
        this.isim = isim;
    }

    public String getIsim() {
        return isim;
    }
}
```

```

package ara2;

public class OgretimUyesi extends Kisi {
    private Integer sicilNo;

    public OgretimUyesi(String isim, Integer sicilNo) {
        super(isim);
        this.sicilNo = sicilNo;
    }

    public Integer getSicilNo() {
        return sicilNo;
    }
}

```

```

package ara2;

public class Ogrenci extends Kisi {
    private Integer ogrenciNo;

    public Ogrenci(String isim, Integer ogrenciNo) {
        super(isim);
        this.ogrenciNo = ogrenciNo;
    }

    public Integer getOgrenciNo() {
        return ogrenciNo;
    }
}

```

**Soru 2:** (20p) GecersizKapasiteException ve OgrenciZatenKayitliException sınıflarını kodlayınız.

```

package ara2;

public class GecersizKapasiteException extends java.io.IOException {
    public GecersizKapasiteException( ) { }
    public GecersizKapasiteException( String mesaj ) { super(mesaj); }
}

```

```

package ara2;

public class OgrenciZatenKayitliException extends java.io.IOException {
    public OgrenciZatenKayitliException ( ) { }
    public OgrenciZatenKayitliException ( String mesaj ) { super(mesaj); }
}

```

**Soru 3:** (10p) Ders sınıfının setKapasite metodunu kodlayınız.

```

    public void setKapasite(int kapasite) throws GecersizKapasiteException {
        if( kapasite >= getKayitliOgrenciSayisi() )
            this.kapasite = kapasite;
        else
            throw new GecersizKapasiteException();
    }

```

**Soru 4:** (50p) USIS sınıfını kodlayınız.

```
package ara2;
import java.util.*;
public class USIS {
    private HashMap<String, Ders> dersler;
    private HashMap<Integer, Kisi> kisiler;

    public USIS( ) {
        dersler = new HashMap<String, Ders>( );
        kisiler = new HashMap<Integer, Kisi>( );
    }

    public void ogrenciOlustur( String isim, Integer ogrenciNo ) {
        Ogrenci ogrenci = new Ogrenci(isim, ogrenciNo);
        kisiler.put(ogrenciNo, ogrenci);
    }

    public void hocaOlustur( String isim, Integer sicilNo ) {
        OgretimUyesi hoca = new OgretimUyesi( isim, sicilNo );
        kisiler.put(sicilNo, hoca);
    }

    public void dersOlustur( String isim, String kod, Integer hocaSicil ) {
        Ders ders = new Ders(kod, isim);
        OgretimUyesi hoca = (OgretimUyesi) kisiler.get(hocaSicil);
        ders.setHoca(hoca);
        dersler.put(kod, ders);
    }

    public void setKapasite( String dersKodu, int kapasite ) {
        Ders ders = dersler.get(dersKodu);
        if( ders == null ) return;
        try {
            ders.setKapasite(kapasite);
        }
        catch (GecersizKapasiteException e) {
            e.printStackTrace();
        }
    }

    public void derseOgrenciEkle( String dersKodu, Integer ogrenciNo ) {
        Ogrenci ogrenci = (Ogrenci) kisiler.get(ogrenciNo);
        Ders ders = dersler.get(dersKodu);
        ders.ogrenciEkle(ogrenci);
    }
}
/*
 * HashMap.values() kullananlardan, işi uzattıkları için 2 puan kırılmıştır.
 * dersOlustur metodunda dersin önceden oluşturulmuş olduğunu düşünüp
 * dersler.get( dersKodu ) ile nesne elde edenlerden hiç puan kırılmamıştır.
 */
```