



Java ile Nesne-Merkezi Programlamaya Giriş

Bölüm 7 - Nesne-Merkezi Programlamaya Giriş



Eğitmen:

Akın Kaldıroğlu

Çevik Yazılım Geliştirme ve Java Uzmanı

Konular



- **Soyutlama**
 - Model
 - Soyutlama ve Programlama
- **Sınıf ve Nesne**

Soyutlama

Soyutlama (Abstraction) - I



- **Soyutlama (abstraction)**, bir şeyin en önemli özelliklerini ön plana çıkarırken, önemli olmayan özelliklerini bastırmaktır, görmezden gelmektir:
- Önemli olan özellikler, genel olarak o şeyi diğer şeylerden ayırt eden unsurlardır ya da ana, **asli özelliklerdir (essential properties)**,
- Ayırt edici olmayanlara ise ikincil (primary) ya da arızı (accidental) özellikler denir.

Soyutlama (Abstraction) - II



- Zihnimiz karşılaştığı her nesneyi tek tek algılamak yerine, nesneleri, karakteristik özellikleriyle algılar, sonra da ya zihinde var olan kavramsal bir kategoriyle örtüştürür ya da böyle bir kategori yoksa, bu nesneden yola çıkarak yeni bir kategori oluşturur.
- Bu anlamda **soyutlama, seçici bilgi ya da cehalettir (selective knowledge or ignorance)**.

Soyutlama (Abstraction) - II



- **soyutlama** İng. **abstraction** (Lat. abstractio < abstrahere = çekip çıkarmak, soymak, ayırmak): 1. Gerçekte ayrılamaz olanı düşüncede ayırma eylemi. (Ör. Biçimi, rengi, boyutları özdekten ayırıp düşünme.) 2. Geneli ve öz olanı arınmış bir biçimde elde etmek için özle ilgili olmayanı bir yana bırakma. Bir tasarımın ya da bir kavramın nitelik ve bağıntı gibi öğelerini göz önüne almayarak dikkati doğrudan doğruya kavrama çeken düşünme eylemi. (Ör. Üçgen kavramında; üçgenin büyük, küçük, eşkenarlı ya da dik açılı oluşunu göz önüne almamak gibi.)
- Bilim ve Sanat Terimleri Sözlüğü, <http://www.tdk.gov.tr/>

Soyutlama (Abstraction) - IV



- Soyutlamanın İngilizce karşılığı **abstraction**'dır.
- Britannica'da şöyle tarif edilmektedir:

The cognitive process of isolating, or "abstracting," a common feature or relationship observed in a number of things, or the product of such a process.

- Soyutlama, bu anlamda **kavramsallaştırma** ile aynı anlamdadır.

Neden Soyutlama? - I



- Gerçeklik/varlık (reality/existence), sonsuz karmaşıklıktadır.
- Bu yüzden zihnimiz onu olduğu gibi kavrayamaz.
- Zihnimiz, soyutlama ile gerçekliği anlayabileceği hale indirger, basitleştirir.
- Dolayısıyla her soyutlama bir basitleştirme, detaylardan arındırma, indirgeme, kavramsallaştırma ve genellemedir.

Neden Soyutlama? - II

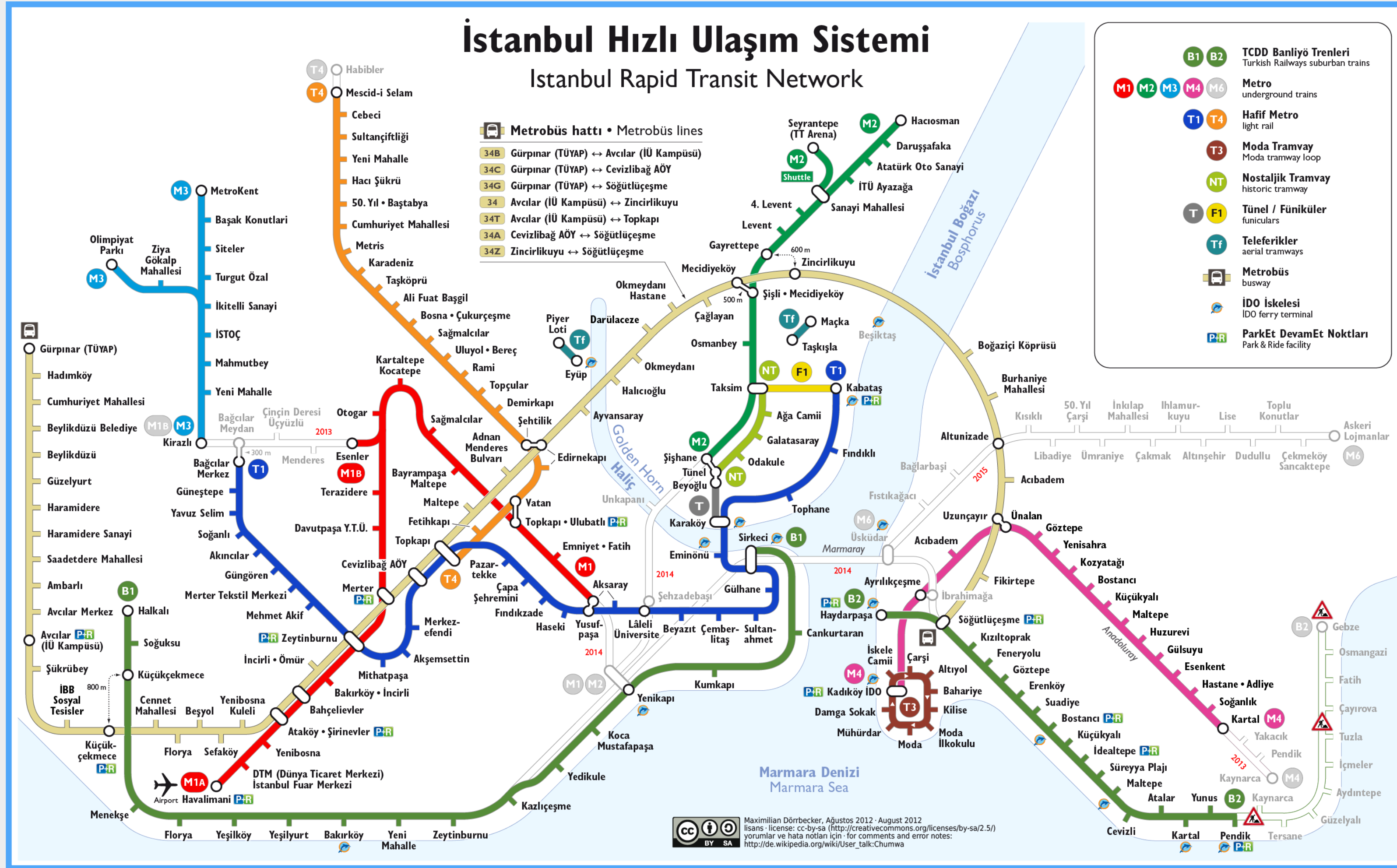


- Zihin detaya girdikçe soyutlamaları genişletir, gerçeğine yakınlaştırır.
- Soyutlamalar, bir bebeğin dünyayı algılama şeklinde apaçık görülür:
 - Köpek algısı, muhtemelen bir bebekte şöyle gelişir.
 - İnsan olmayan canlı,
 - 4 ayaklı ve kuyruğunu sallayan canlı,
 - 4 ayaklı, canayakın, havlayan ve kuyruğunu sallayan canlı.



- Haritalar, soyutlamaya mükemmel bir örnektirler:
 - Farklı türdeki haritalar, aynı gerçekliğin farklı yönlerini betimlerler, amaçları dışındaki özellikleri göstermezler.
- Amaç ya da ilgi alanı, aynı gerçekliğin farklı soyutlamalarına götürür:
 - Otel zincirlerinin ya da benzin istasyonlarının haritaları,
 - Aynı binaya bakan farklı meslekteki kişilerin farklı zihinsel tasarımları.

Haritalar - II





- Soyutlamalar, bir şeyin “**ne**”liği (**what-ness**)üzerine yoğunlaşır, “**nasıl**”lık (**how-ness**) üzerine eğilmez.
- Yani soyutlamalar, şeylerin temel özelliklerini, o özelliklerin oluşumundan, nasıl meydana geldiğinden bağımsız olarak ele alır:
 - Araba sürmek için motor yapısını bilmeye gerek yoktur,
 - Ya da bir insanla ortaklık kurmak için onun DNA dizilerini bilmeye de gerek yoktur.

Soyutlama ve Gerçeklik - I



- Soyutlama, sonsuz karmaşıklığı basitleştirerek anlama çabasıdır.
- Bütün soyutlamalar birer kurgudur (construction), insanın gerçekliğe giydirdiği bir kisvedir, çoğu gerçeklikte ya yoktur ya da bizim algıladığımız şekilde değildir:
- Banka ya da banka hesabı ya da bilgisayarda dosya sistemi,
 - Dosya sistemine karşın hard diskin fiziksel yapısını ve dosya kopyalamayı düşünün.
- Metre, gram vb. ölçüler.

Soyutlama ve Gerçeklik - II



- Dolayısıyla insan zihni, gerçekliği algılamak için olduğu kadar kendi gerçekliğini oluşturmak için de soyutlamalar yapar.



sselsoft
build better, deliver faster

Model



- Nesnelerin çoğu zaman tek bir açıdan yapılan soyutlamayla anlaşılması zordur.
- Bir şey ile ilgili farklı açılardan ya da ilgi alanlarından yapılan soyutlamaların bütününe **model** denir.
- Modeller, bir nesneyi daha bütünsel olarak ifade ederler, çünkü birden fazla soyutlama barındırırlar.
- Dolayısıyla zihnimiz, modellerle gerçekliğin daha rahat algılaması amacıyla, birden fazla bakış açısıyla elde edilmiş soyutlamalarını yapar.



- Başka bir deyişle soyutlamaları da sınıflandırırız:
 - Bir binanın statik, dinamik açılardan özelliklerini ve davranışlarını farklı soyutlamalarla ortaya koyarız.
 - Aynı binanın elektrik donanımı, ısıtma/soğutma/havalandırma düzeni ve ısı yalıtımı vs., hepsi birer soyutlamadır.
- Bir binayı ancak bu soyutlamaların bütünü temsil edebilir ki buna da binanın modeli denir.

Modellemenin Özellikleri - I



- Sınırı yoktur, sonsuz sayıda ve derinlikte modeller oluşturulabilir.
- Detay miktarı, amaç ve imkanlarla belirlenir.
- Mutlak doğru model ancak gerçekliğin kendisidir.
- Doğruluk ya da yanlışlık, ancak gerçekliğe uygunlukla ölçülebilir,
- Gerekli özellikleri alıp gereksizleri bırakmak, soyutlamanın dolayısıyla da modelin doğruluğunu belirler.

Modellemenin Özellikleri - II



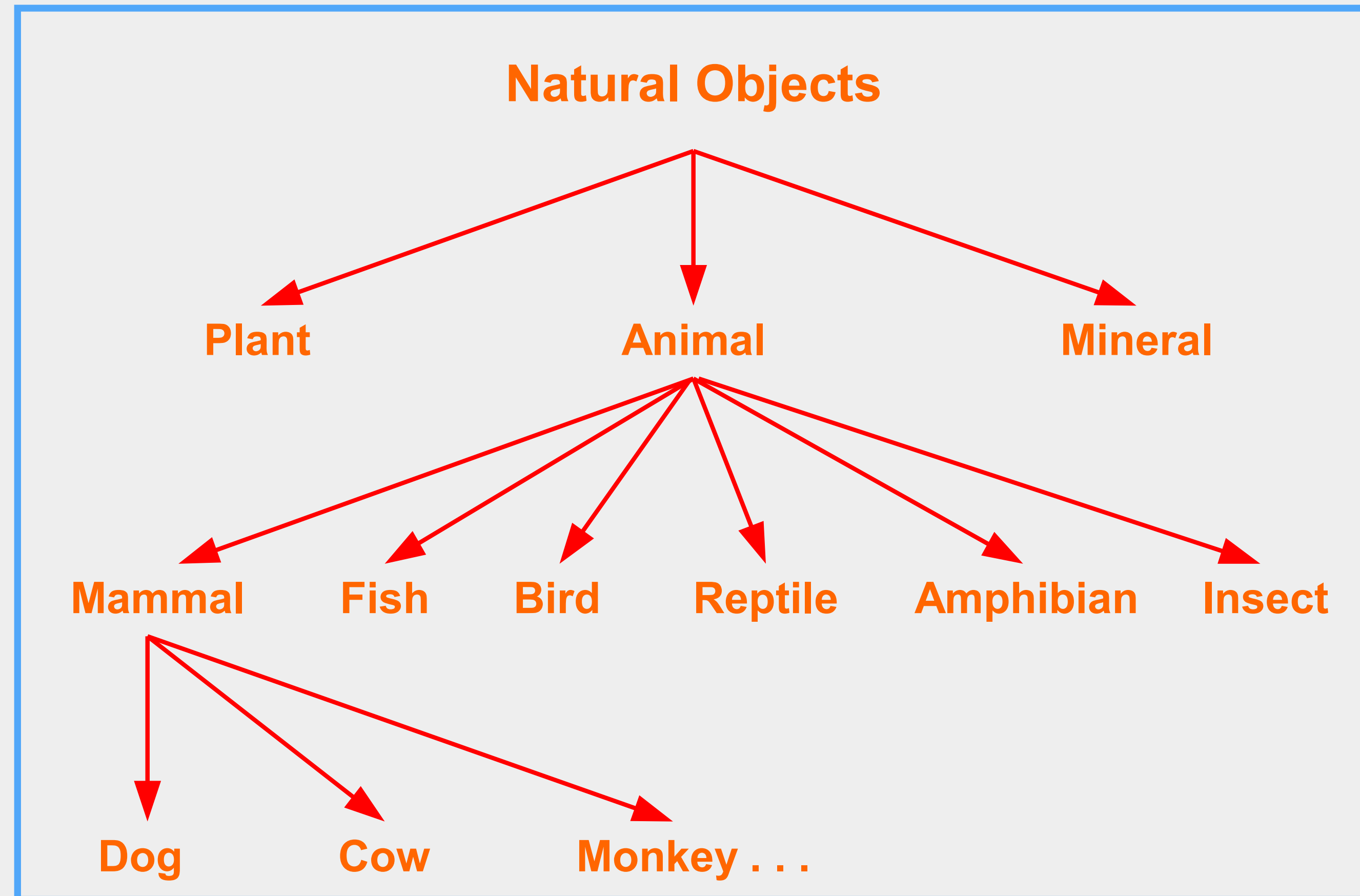
- Modellerin testini yapmak zordur.
- En iyi test, modelin, ilgililerini ne kadar memnun ettiğidir.

Sınıflandırma (Classification)



- Zihin, sonsuz sayıda soyutlama ile uğraşır,
- Ve bu soyutlamalarla elde edilen genellemeler zihinde birer kategori oluşturur,
- Soyutlama ile kategori oluşturmaya, **kategorizasyon (categorization)** ya da **sınıflandırma (classification)** denir.
- Böylece nesneler, olgular, duygular vs. hepsi zihnimizde belli kategoril altına düşürülür ve sınıflara ait hale gelirler.

Tabiattaki Nesnelerin Sınıflandırılması





build better, deliver faster

Soyutlama ve Programlama

Soyutlama ve Programlama



- Peki bütün bunların programlamayla ne ilgisi var?
- Programlama karmaşık soyutlamardan oluşan modeller kurmadır.
- Programlama dilleri de çok farklı soyutlamaları barındıran modellerdir:
 - Değişkenler ve veri tipleri
 - Operatörler ve kontrol yapıları
 - Fonksiyonlar
 - Sınıflar, vs.

Sinif ve Nesne



- “*Köpek dediğin sadık olmalı.*” ya da “*Köpekler çok sadık hayvanlardır.*” cümlelerinde, kavramsal, soyut olarak köpek sınıfı, cinsi kastedilir ve söylenen, var olmuş ve olacak bütün köpekler için geçerlidir.
- “*Komşunun köpeği çok tatlı.*” cümlesinde ise köpek sınıfının bir örneği ya da nesnesi, somut olan bir canlıdan bahsediyoruz demektir.
- Köpek, ilk durumda bir sınıfı, ikinci durumda ise bir nesneyi temsil eder.

Sınıf ve Nesne - II



- Bu anlamda sınıf (class), soyut olanı (abstract), tümeli (universal), örneklerinin hepsine genel (general) olanı temsil eder.
- Nesne ise, somut olanı (concrete), yani soyut olanın ete kemiğe bürünmüş halini, tikeli (particular), bir örneği (instance) temsil eder.

Nesne (Object) - I



- **Nesne (object)**, insan zihninin algıladığı ve özellikleri olan herhangi bir kavramsal ya da fiziksel şeydir:
 - Öğrenciler, derslere devam ediyorlar.
 - Öğretmen, sınıfta öğrencileri dinliyor.
 - Dersler yarın başlıyor.

Nesne (Object) - II



- Nesnelerin özellikleri vardır ve bu özellikler, nesnelerin **durumunu (state)** tarif ederler.
- Ayrıca nesnelerin **davranışları (behavior)** vardır:
 - Sarı boyalı sınıfta öğrenci şiir okuyor.
 - Kırmızı top suya yuvarlandı.
- Dolayısıyla nesneler, bilirler ve yaparlar.

Sınıf (Class) - I



- Sınıf, nesneler için bir kalıptır, şablondur, yani kendisinden üretilecek olan nesnelerin sahip olacağı özellikler ile davranışları tarif eder.
- Sınıf, kendisinden türetilecek olan nesnelerinin
 - özelliklerini değişik tiplerde değişkenlerle (variables),
 - davranışlarını ise metotlarla (method) (fonksiyon (function)/prosedür (procedure)) ifade eder.

Sınıf (Class) - II



- Nesnenin özelliklerinin bütününe **durum (state)**, metotların bütününe de **arayüz (interface)** denir.
- Durum bilgisi ve arayüz, sınıfta tanımlanır, sınıftan üretilen nesneler ise durum bilgisine sahip olurlar, arayüzle tarif edilen davranışları yerine getirirler.

Sınıf (Class) - III



- Aynı sınıftan üretilen nesneler aynı tipte olurlar:
 - Aynı özelliklere sahiptir ama özelliklerin değerleri değişebilir,
 - Aynı davranışlara sahiptir,
- Davranışlar genelde duruma bağlı olduğundan, farklı durumdaki nesnelerin davranışları da farklı olur.



- Yazılımın nesnesi ise hayatımızdaki kavramsal ya da fiziksel bir nesneyi temsil etmek üzere, onun özelliklerini ve davranışlarını ifade eden yapıdır:
- Yazılımın nesnesi, temsil ettiği gerçek dünyadaki nesnenin durumunu, sınıfında tanımlanan değişkenlerle, davranışlarını da metotlarla yerine getirir.

Durum (State)



- Nesnelerin **durumu** (**state**) ile daha çok durağan (static) özellikleri kastedilir ve programlama dillerinde farklı tiplerde bir grup değişken ile ifade edilir.
- Nesnenin durumunu oluşturan her bir ayırık bilgiye ise **özellik** (**attribute, property**) denir:
 - Öğrenci: No, isim, soy isim, doğum tarihi, cinsiyet, adres, bölüm, aldığı dersler, vs.
 - Ders: No, isim, bölüm, veren kişi, kredi sayısı, vs.

Davranış (Behavior)



- Nesneler davranırlar, belli işleri yerine getirirler.
- Yazılım nesnelerinin davranışlarına, yerine getirdiği **sorumluluk (responsibility)**, verdiği **hizmet (service)** ya da aldığı **mesaj (message)** olarak bakmak, işimizi kolaylaştırır:
- Öğrenci: Kayıt olur, ders alır, sınava girer, vs.
- Ders: Öğrencinin kaydolmasına/bırakmasına izin verir, ön şart dersleri hakkında bilgi verir, vs.

Mesajlaşma (Messaging) - I



- Bir nesne, başka bir nesneye karşı şu 3 şekilden biri ile bir sorumluluk yerine getirir:
 - Nesne, diğer nesnelere kendi durumu hakkında bilgi verir,
 - Nesne, diğer nesnelerden gelen istek üzerine kendi durumunu değiştirir,
 - Nesne, başka bir nesnenin isteği üzerine, bir işi ya da faaliyeti yerine getirir ve muhtemelen bu faaliyet sonucunda isteyen nesnelere bilgi geri döndürür.

Mesajlaşma (Messaging) - II



- Nesne, yukarıdaki işleri yaparken, özellikle 3. tür işlerde, muhtemelen başka nesnelerden yardım alır.
- Nesnelerin bu şekilde birbirlerine hizmet vermesine **mesajlaşma (messaging)** denir.



- Nesne-merkezli yazılım sistemi (object-oriented software system) ise, birbirleriyle mesajlaşan ve bu şekilde iş süreçlerini yerine getiren bir grup nesneden başka birşey değildir.
- Nesneler, yazılım sisteminin yerine getireceği sorumlulukları paylaşırlar öyle ki her bir nesne, temsil ettiği kavramla ilgili sorumlulukları yerine getirir.



- Bir “Araba” soyutlaması yapalım.
- Soyutlamada bulunması gereken durum bilgileri nelerdir?
- Soyutlamada bulunması gereken davranışlar nelerdir?



- Araba soyutlamasında bulunması gereken davranışlar nelerdir?
 - Gitmek, durmak, hızlanmak, yavaşlamak, vs.
- Araba soyutlamasında bulunması gereken durum bilgileri nelerdir?
 - Marka, model, üretim yılı, hızı, vs.



```
public class Car{
    public String make;
    public String model;
    public String year;
    public int distance;
    public int speed;

    public double go(int newDistance) {
        distance += newDistance;
        double period = (double) newDistance / speed;
        return period;
    }

    public void accelerate(int newSpeed) {
        speed = newSpeed;
    }

    public void stop() {
        speed = 0;
    }

    public String getInfo() {
        String info = "Car Info: " + year + " " + make + " " + model +
            ". Distance: " + distance + " km. and
            traveling at " + speed + " kmph.";
        return info;
    }
}
```

Car



- `org.javaturk.oopj.ch07.Car`



- Az önce yaptığınız Araba soyutlamasını nasıl geliştirebilirsiniz?
- Farklı meslekler nasıl Araba soyutlamalarına sahiptirler?

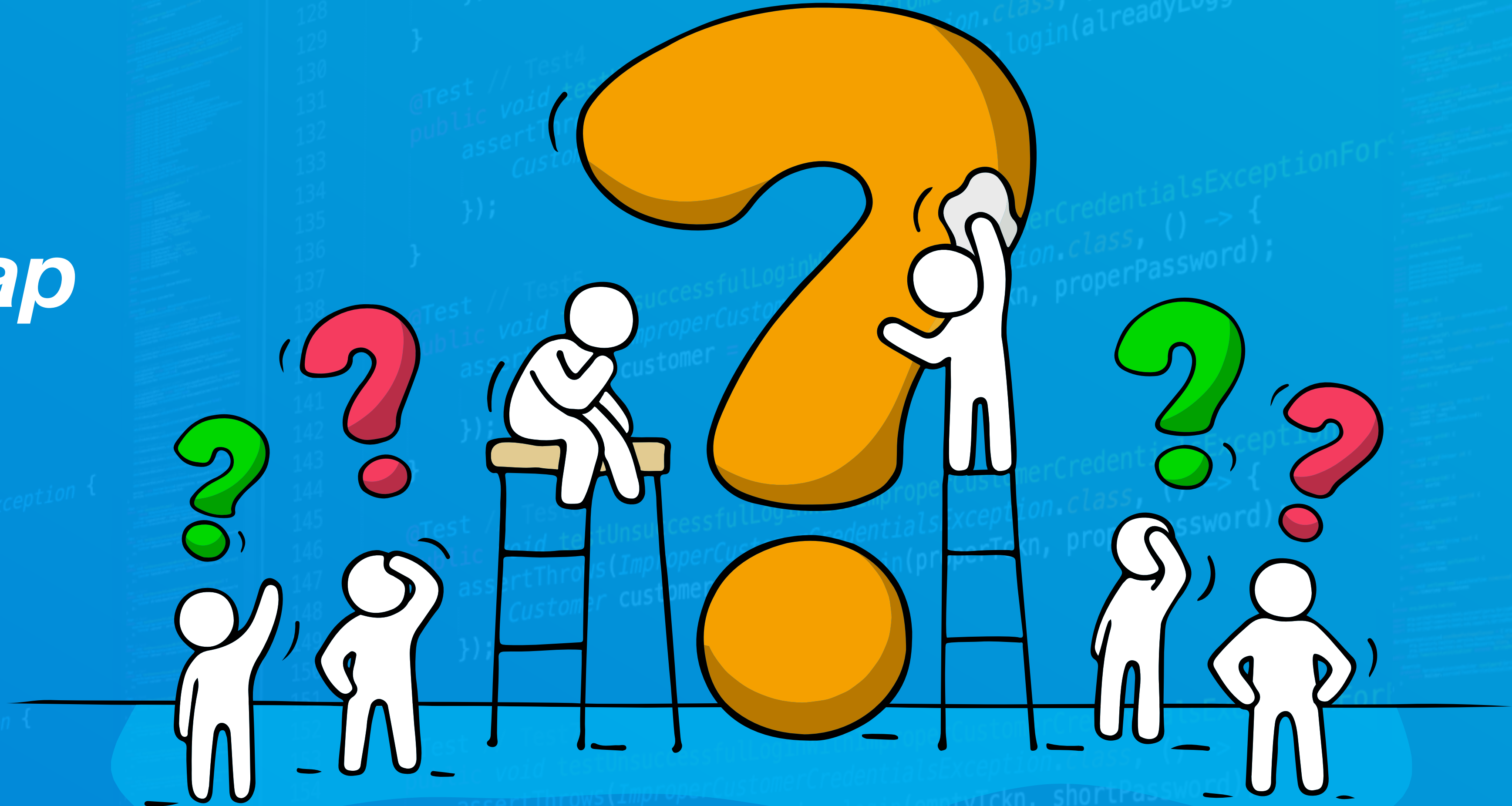


- Aşağıdaki açılardan baktığınızda soyutlamanızda neler değişir?
 - Sürücü
 - Akaryakıt kullanımı, maksimum hız
 - Satıcı
 - Fiyatı
 - Araba tasarımcısı
 - Üstü açılabilir mi?



- Tamirci
 - Motor özellikleri
- Sigortacı
 - Nerede ve kim kullanıyor,
 - Hasarı var mı?
 - Şu anki değeri nedir?

Soru ve Cevap Zamanı!





build better, deliver faster

Ödevler



1. Bir karmaşık sayı (complex number) soyutlamasını nasıl yaparsınız?
 - a. Bir karmaşık sayının durumunu ifade etmek üzere hangi tür bilgileri kullanırsınız?
 - b. Bir karmaşık sayıdan ne isteyebilirsiniz?
 - c. Bu soyutlamayı sınıf olarak ifade edin ve test sınıfında nesnelerini oluşturup üzerinde metot çağrıları yapın.



2. Bir çalışan (employee) soyutlamasını nasıl yapardınız?

- a. Bir çalışanın durumunu ifade etmek üzere hangi tür bilgileri kullanırsınız?
- b. Bir çalışandan ne isteyebilirsiniz?
- c. Bu soyutlamayı sınıf olarak ifade edin ve test sınıfında nesnelerini oluşturup üzerinde metot çağrıları yapın.



3. Bir kahve makinası (coffee machine) soyutlamasını nasıl yapardınız?
- a. Böyle bir makinanın durumunu ifade etmek üzere hangi tür bilgileri kullanırsınız?
 - b. Bu makinadan ne isteyebilirsiniz?
 - c. Bu soyutlamayı sınıf olarak ifade edin ve test sınıfında nesnelerini oluşturup üzerinde metot çağrıları yapın.



4. Bir aşk (love) soyutlamasını nasıl yapardınız?
 - a. Bir aşkın durumunu ifade etmek üzere hangi tür bilgileri kullanırsınız?
 - b. Bir aşk nesnesinden ne isteyebilirsiniz?
 - c. Bu soyutlamayı sınıf olarak ifade edin ve test sınıfında nesnelerini oluşturup üzerinde metot çağrıları yapın.

Bölüm Sonu

Soru ve Cevap Zamanı!

