
دوره برنامه نویسی C++ پیشرفته

دانشگاه ولی عصر رفسنجان

مهندس حسین بازماندگان

جلسه پنجم

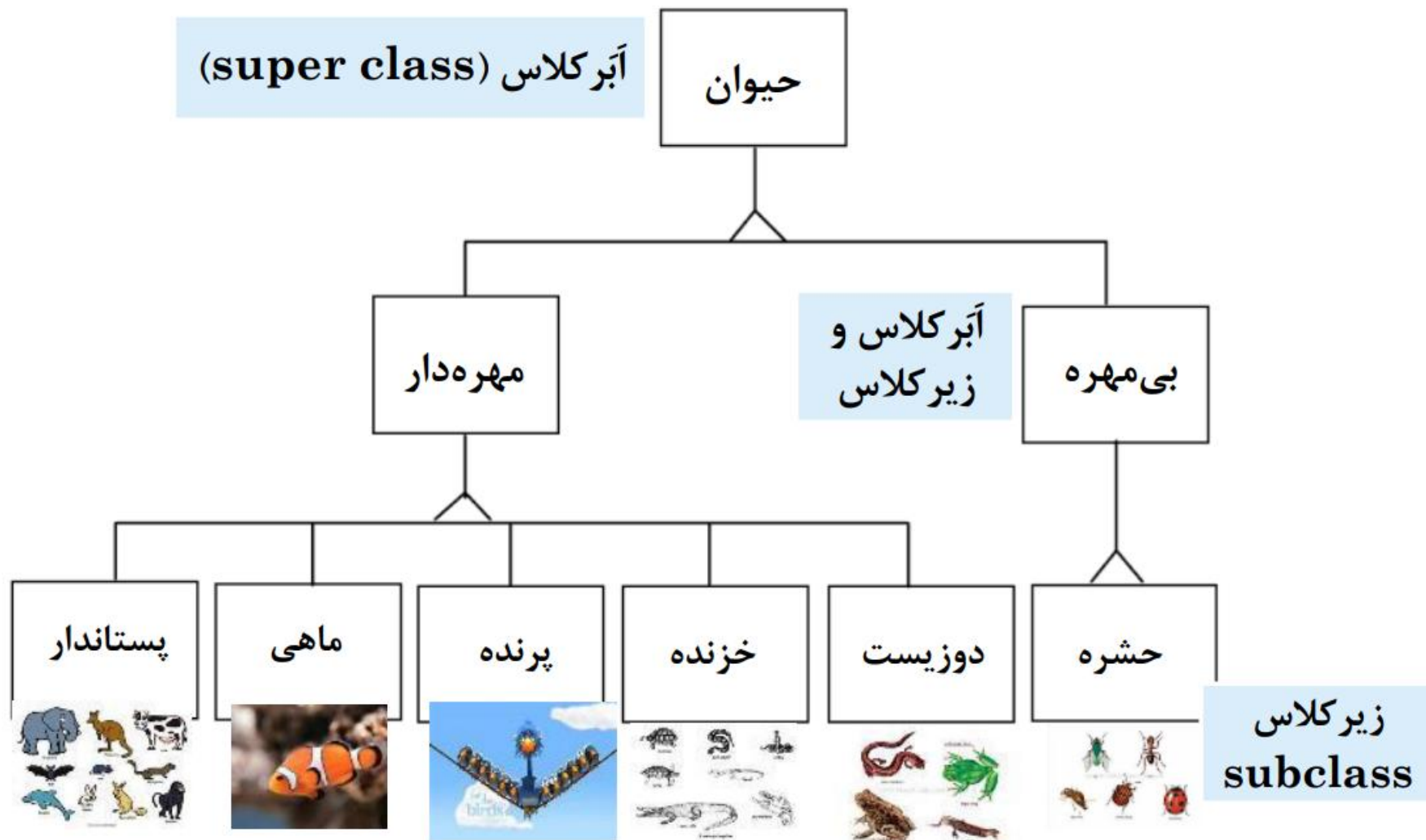
ارث بری (Inheritance) بخش اول

سرفصل مطالب

- آشنایی با مفهوم وراثت
- نحوه پیاده سازی زیرکلاس ها
- نمایش وراثت در UML Class Diagram

آشنایی با مفهوم وراثت

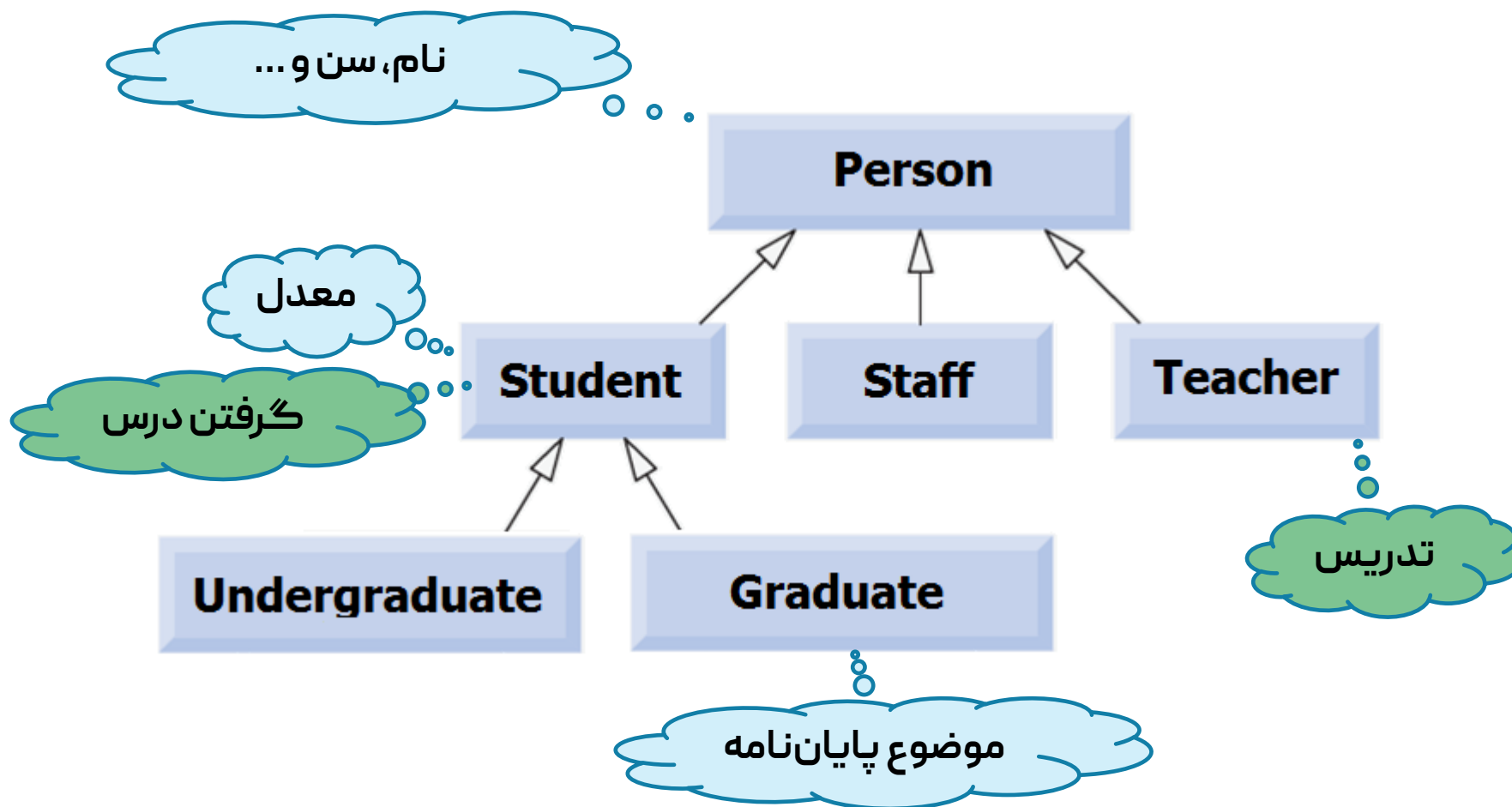
سلسله مراتب کلاس ها



معنای سلسله مراتب انواع کلاس‌ها

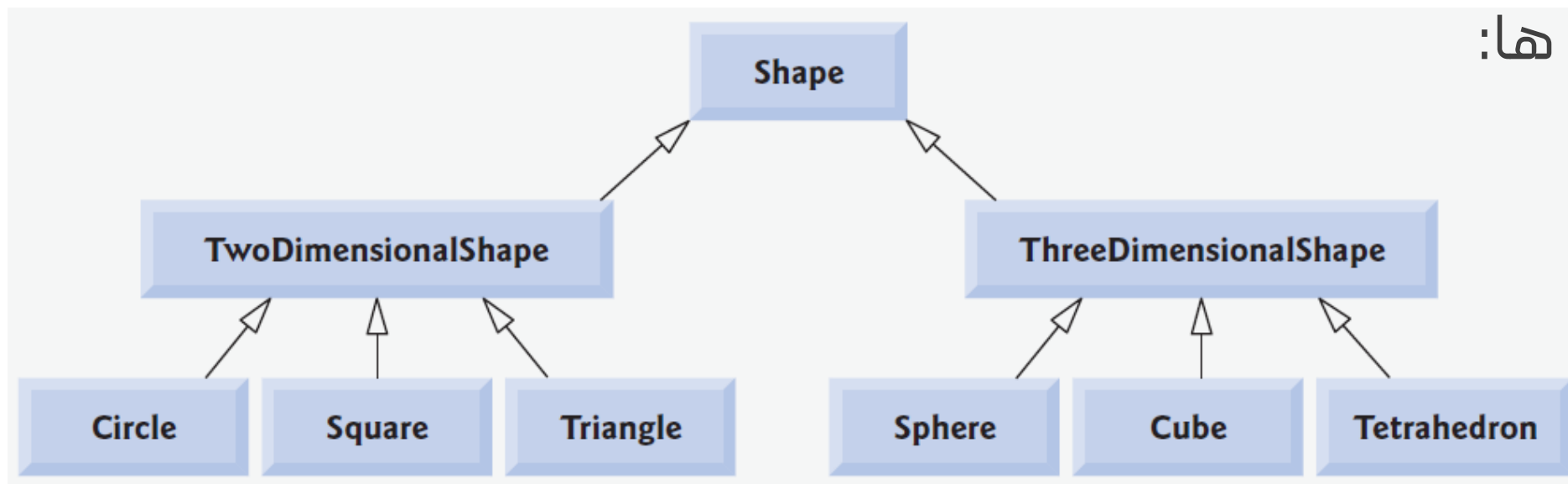
- دایره نمونه‌ها (اشیاء) در زیرکلاس محدودتر می‌شود
- زیرکلاس، ویژگی‌ها و رفتار آبَرکلاس را به ارث می‌برد
- اصطلاح وراثت و ارث‌بری (Inheritance)
- مثال: هر حیوان، ویژگی‌هایی مانند «سن» و «وضعیت سلامتی» دارد
- این ویژگی‌ها به همه زیرکلاس‌ها به ارث می‌رسد
- همه زیرکلاس‌های مستقیم و غیرمستقیم: مهره‌دار، بی‌مهره، ماهی، حشره و ...
- یعنی هر شیء از زیرکلاس‌ها هم همین ویژگی‌ها را دارد
- احتمالاً ویژگی‌های دیگری هم دارد (مثلاً هر ماهی «سرعت شنا کردن» دارد)
- مثال: هر حیوان، رفتارهایی مانند «غذا خوردن» و «جابجا شدن» دارد
- پس همه زیرکلاس‌ها هم این رفتارها را دارند (این رفتارها را به ارث می‌برند)

سلسله مراتب کلاس‌ها (مثال)

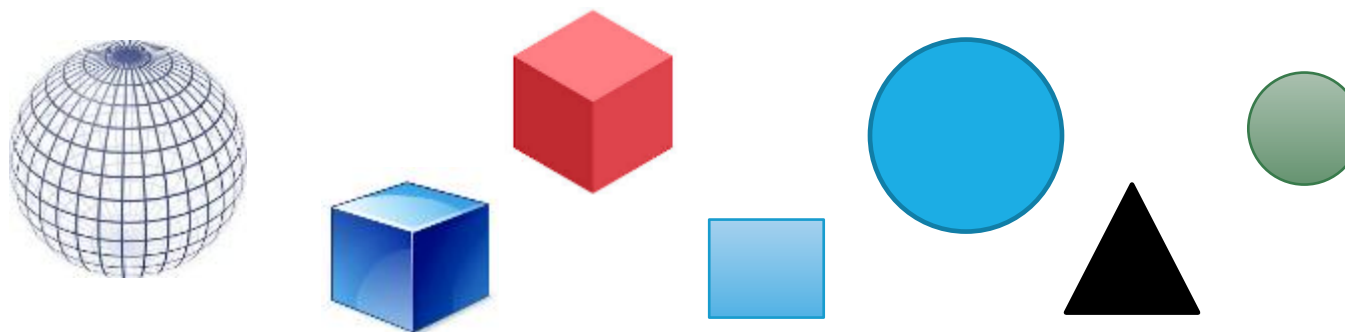


سلسله مراتب کلاس ها (مثال)

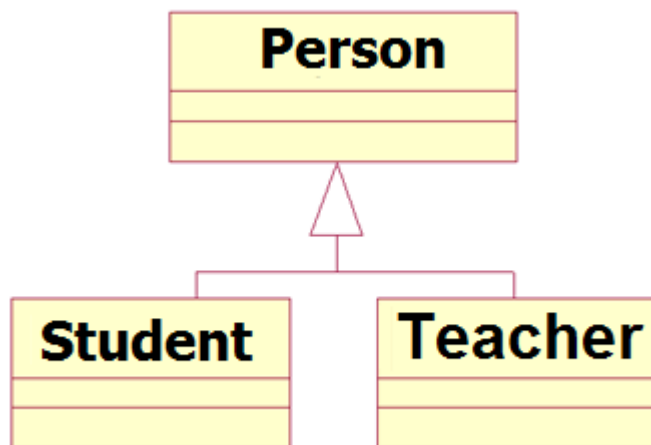
• کلاس ها:



• نمونه ها:



انواع عام‌تر و انواع خاص‌تر



- آبَرکلاس، نوع عام‌تری از زیرکلاس است
- زیرکلاس، نوع خاص‌تری از آبَرکلاس است
- تأکید: زیرکلاس و ابرکلاس هر دو «کلاس» هستند
- هر شیء از زیرکلاس، شیئی از ابرکلاس هم هست
- دانشجو زیرکلاس انسان است
- دانشجو نوع خاص‌تری از کلاس انسان است
- (دایره محدودتری از نمونه‌ها را شامل می‌شود)
- همه ویژگی‌ها و رفتارهای انسان در دانشجو هم وجود دارد
- مثل: نام، سن، غذا خوردن و ... البته دانشجو ویژگی‌ها و رفتارهای دیگر هم دارد
- علی‌علوی یک دانشجو است (یک نمونه، شیء). پس علی‌علوی، انسان هم هست

واژه‌شناسی

- کلاس اصلی:
- کلاس پایه (Base Class)
- آبَرکلاس (Superclass)
- کلاس والد (Parent Class)
- کلاس وارث:
- کلاس مشتق (Derived Class)
- زیرکلاس (Subclass)
- کلاس فرزند (Child Class)

نحوه پیاده سازی زیرکلاس ها

فرض کنید ارث بری نداشته‌ایم

کلاس Animal

```
class Animal {  
public:  
    void eat() {  
        cout << "This animal eats food." << endl;  
    }  
};
```

کلاس Dog

```
class Dog {  
public:  
    void eat() {  
        cout << "This animal eats food." << endl;  
    }  
    void bark() {  
        cout << "Woof! Woof!" << endl;  
    }  
};
```

با کمک ارث بری

کلاس Dog

```
class Dog: public Animal {  
public:  
    void bark() {  
        cout << "Woof! Woof!" << endl;  
    }  
};
```

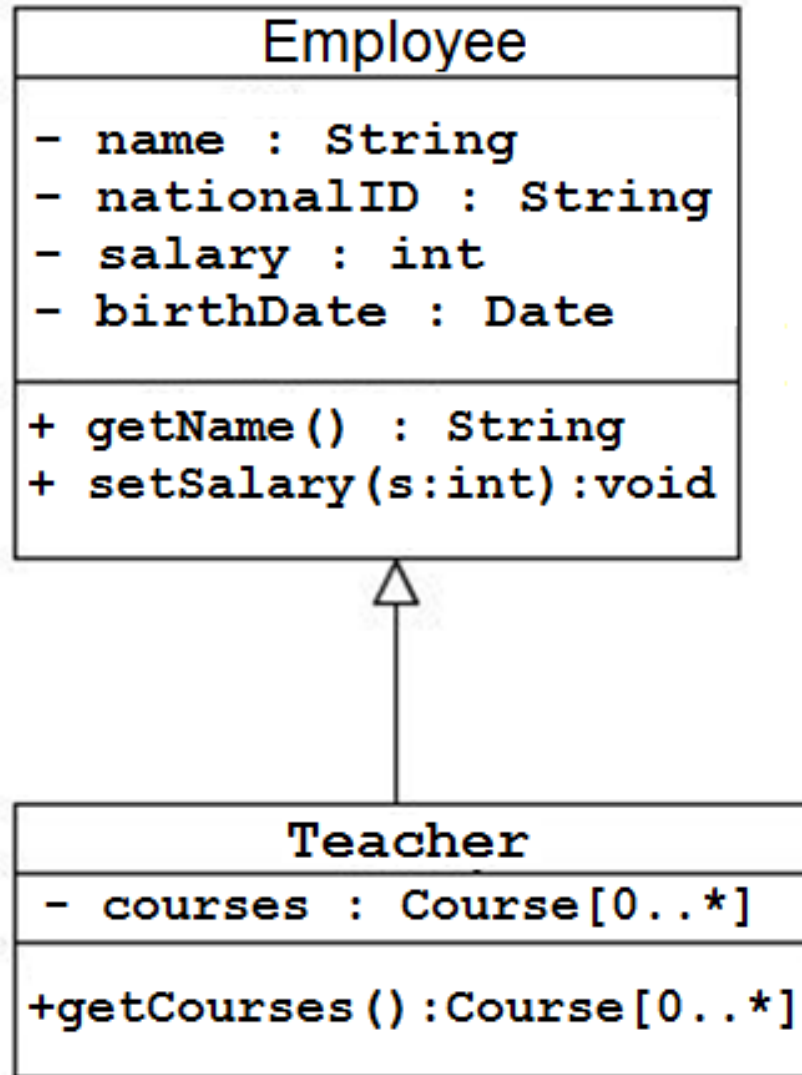

وراثت در زبان‌های شیء‌گرا

- زبان‌های برنامه‌نویسی شیء‌گرا تعریف وراثت را ممکن می‌کنند
 - از جمله سی‌پلاس‌پلاس
- از این امکان برای تعریف زیرکلاس‌ها استفاده می‌شود
 - بدون این که نیازی به کپی‌کردن از آب‌رکلاس باشد
- وراثت یکی از راه‌های استفاده مجدد از کد است
 - `code reuse`
- کدی که در آب‌رکلاس نوشته شده، در زیرکلاس بازاستفاده می‌شود
 - دوباره نوشته نمی‌شود



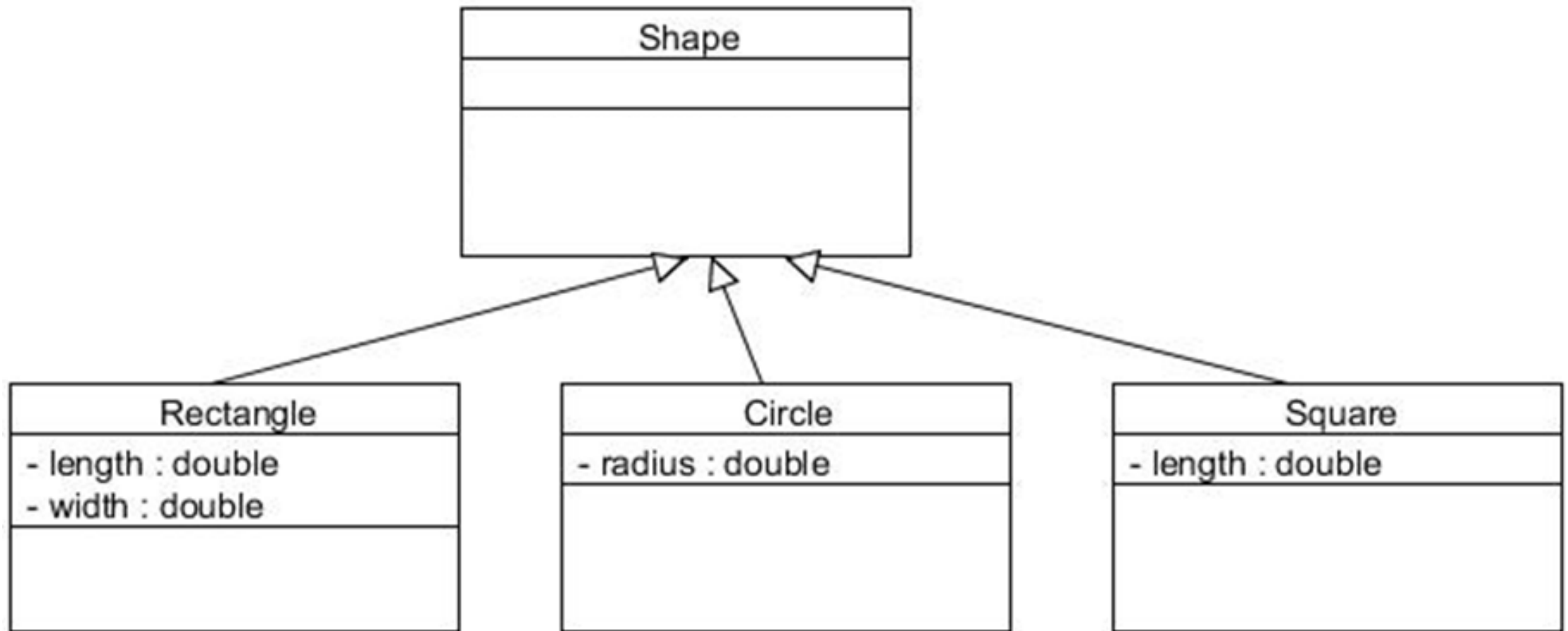
UML class diagram

نمودار UML برای کلاس‌ها



- UML Class Diagram
- نموداری برای توصیف طراحی کلاس‌ها
- کاربردهای مختلفی دارد
- مثال: تعامل بین طراح و برنامه‌نویس
- نمودار UML قواعد خاصی دارد
- مخصوص زبان جاوا نیست
- نمودار UML شامل:
 - متدها و ویژگی‌های کلاس‌ها
 - سطوح دسترسی
 - روابط بین کلاس‌ها
 - (وراثت: یکی از انواع رابطه ممکن است)

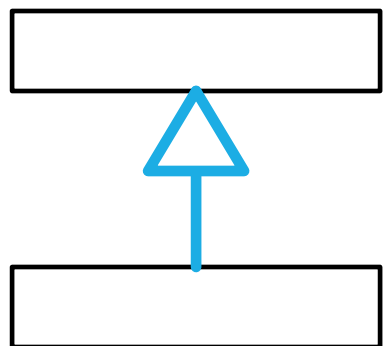
مثال دیگری برای UML class diagram



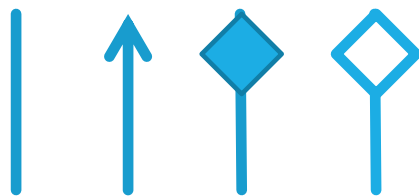
رابطه ISA

- اصطلاحاً: بین زیرکلاس و اَبَرکلاس رابطه ISA برقرار است

- Rectangle **is a** Shape
- A rectangle instance **is a** Shape instance too



- رابطه وراثت (ISA) در UML با یک فلش با سر مثلث تو خالی نمایش داده می شود:



- شکل های دیگر معانی دیگری دارند:

مرور یک مثال از کاربرد وراثت