Kilian Lipinsky
7421772
klipinsk

Department Mathematik/Informatik, Abteilung Informatik
Software & System Engineering
Weiterführende Konzepte der Programmierung, SS 2025

Niklas Bockholt, Chiara Hafner, Alicja Bezpalko,
Jiyan Alis, Dilara Günay,
Simon Wolf, Adrian Bajraktari, Dr. Mersedeh Sadeghi

UNIVERSITÄT
ZU KÖLN

Homework 3. Object-Oriented Programming

Registration Deadline: 27.04.2025, 23:59
Hand-in Deadline: 30.04.2025, 23:59

Exercise 1. Who Can See What?

Create a class `Account` with the following attributes:

- `public String owner`.
- `private double balance`.
- `protected int pin`.
- `String internalNote` (default / package-private access).

Write methods to:

- return the balance (`getBalance()`)
- change the pin (only with correct current pin)

Then create:

- class `AccountManager` in the **same** package.
- class `ExternalAudit` in a **different** package.
- class `CompanyAccount` in a different package that **inherits from** `Account`.

Try accessing all four attributes from all three classes.

- Explain which accesses fail and why.
- Suggest alternative designs (e.g., getters).
- What is the difference between using `protected` vs. using `private` and providing a getter for subclasses to access an instance variable?

Exercise 2. Who Are You Really?

Create:

- class `Message` with method `getType()` which returns `"Generic"`.
- class `Email extends Message` where `getType()` returns `"Email"`, and a method `send()` that prints `"E-Mail sent"`.

- class `SMS extends Message` where `getType()` returns `"SMS"`.

Write:

```
Message m1 = new Email();
Message m2 = new SMS();

System.out.println(m1.getType());
System.out.println(m2.getType());
// m1.send(); // Uncomment this line and explain what happens
```

Answer the following questions.

- What is the static type of `m1`? What is its dynamic type?
- Which method is called? Why?
- Why can't `send()` be called directly?
- How could you safely call `send()` anyway (e.g. `instanceof`)?

Nr.1

Accessing Attributes from the Classes

Account Manager

- owner is accessible
- InternalNote is auch
- Balance und pin nicht, da private und protected

ExternalAudit

- Owner is accessible, da public
- Balance nicht, da private
- Pin nicht da protected und nicht subclass von Account
- InternalNote: nicht accessible da es package private

CompanyAccount

- Owner public
- Pin: accessible da protected und CompanyAccount ist subclass von Account
- Balance: not accessible da package private und Company Account in anderem Package

Nr.2

Der Code wird nicht funktionieren, er versucht die Methode send() aufzurufen, allerdings ist die Message m1 vom Typ Email, diese Klasse hat aber keine MEthode send() somit ist es nicht kompilierbar. Der static Typ ist Message, der dynamic ist Email.

Man könnte es umgehen indem man instanceof verwendet, dadurch wird geprüft, ob das Objekt wirklich vom Typ Email ist, der cast (Email) m1 sagt, dem Compiler, dass die Message ein Email Objekt ist und so wird die Send Methode zugänglich

Git: https://github.com/Lilli-jr/PK2Homework.git