

Задача по Java: открываем портал в ад!

Денис Гладкий
denis.gladkiy@yahoo.com

Управление prepaid карт, ГК ЦФТ

13 сентября 2013 г.

Всех с днём программиста! Вот вам развлекашка. Вчера с Александром Казаниным мы открыли портал в ад. Теперь предлагается и вам это сделать. Итак, есть код:

```
interface NewRunnable {
    void buz ();
}

abstract class CrazyDevice {
    private final NewRunnable delegate;
    CrazyDevice(final NewRunnable code) {
        assert null != code;
        delegate = code;
    }
    void work() {
        delegate.buz ();
    }
}

class HellPortal extends CrazyDevice {
    private final String name;
    HellPortal(final String name) {
        super(new NewRunnable() {
            @Override public void buz () {
                //can not reference 'open' before supertype constructor
                open ();
            }
        });
        this.name = name;
    }
    void open () {
        System.out.println("The_portal_' + name + "'_is_opened.");
    }
}
```

Открыть портал в методе «`buz`» не выходит: компилятор нам вежливо сообщает, что нельзя обращаться к членам класса-наследника до того, как отработал конструктор базового класса. Оно и понятно — получим окровавленную ногу и кучу неинициализированных полей. Задача: переделать код так, что б всё-таки портал открывался. Ограничения:

- Сохранить создание экземпляра «`NewRunnable`», как объекта анонимного класса, передаваемого в конструктор базового класса «`CrazyDevice`».
- В методе «`buz`» данного экземпляра, должен вызываться метод «`open`» объекта класса «`HellPortal`». Этот объект должен быть тем самым, у которого вызывается конструктор.
- Код должен компилироваться без предупреждений (warnings) и, естественно, не должен содержать их подавлений (например, аннотаций `@SuppressWarnings`).
- Использовать механизм отражения (reflection: класс `Class` и пр.) запрещается.
- Код не должен содержать ни одного приведения типов.
- «`HellPortal`» должен по прежнему наследоваться от «`CrazyDevice`».
- Метод «`open`» не должен статическим (static).