

Subclasses and inheritance Labs

PGL-1. Drawing objects II

Lesson Page 4-1

Introduction

This Lab gives you practice in drawing instances of subclasses. You won't need a computer, but get out a few sheets of paper.

Step 1. Draw an object

Consider the following classes:

```
public class Ex {
    public static final int ZERO= 0;
    private int h;

    public Ex(int ph) {
        h= ph;
    }

    public int getH() {
        return h;
    }

    public String toString() { ... }

    public static int what(int x) { ... }
}

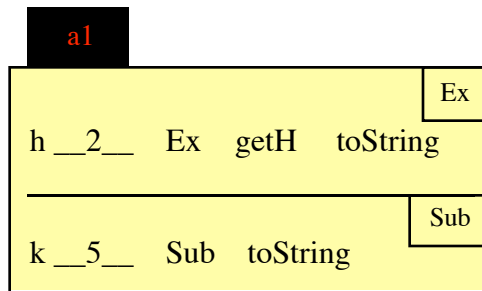
public class Sub extends Ex {
    private int k;

    public Sub(int pk) {
        k= pk;
    }

    public String toString() { ... }
}
```

Below is an instance of class `Sub`. Below the horizontal line are all the instance variables and methods that are declared in `Sub`; above the line are all the instance variables and methods that are declared in superclass `Ex`. Variables `h` and `k` have been given arbitrary values.

Subclasses and inheritance Labs



On your paper, draw an instance of class `Ex` and another instance of class `Sub`.

Step 2. More subclasses

Consider also these two subclasses of class `Sub`:

```
public class SubSub1 extends Sub {
    public SubSub1() {
        super(5);
    }

    public int hPlus1() {
        return getH()+1;
    }

    public String toString() { ... }
}

public class SubSub2 extends Sub {
    private int p;
}
```

Draw an instance of class `SubSub1` and an instance of class `SubSub2`.

Step 3. Taking class `Object` into account

Class `Object` is automatically the superclass of all classes that don't explicitly extend a class. For our purposes, class `Object` defines two instance methods: `equals` and `toString`.

Redraw all the objects that you drew so far (on another sheet of paper), showing explicitly the contribution of class `Object` to each object. If you do not know what to do, listen to the last activity on lesson page 4-2.