CSE1710

Week 09, Lecture 17

Fall 2013 ◆ Thursday, Nov 07, 2013

YORK UNIVERSITÉ UNIVERSITY

Big Picture

This is the final lecture covering Chapter 4 this week.

Complete your reading of Ch 5 for Thursday, Nov 14th, please.

Reminder

On Tuesday, Nov 12, we will have a term test that **focuses on Ch 4**.

what to study?

- Complete exercises and review questions at the end of the chapter.
- Do you have the skills described in the L16 slides?

YORK UNIVERSITÉ UNIVERSITY

2

RQ 4.3

- Name one way constructors are similar to methods;
  Name two ways constructors are different from methods

YORK UNIVERSITÉ UNIVERSITY

RQ

- Consider the following terms:
  "instance of a class"
  "object"
  What is the difference, if any, between these two
- Consider the following terms:
  "object"
  "object reference"
  What is the difference, if any, between these two?

YORK UNIVERSITÉ UNIVERSITY

RQ's

- When an instance is created of a class, must its reference be stored in a variable?

  For instance, what about these statements…

```
Fraction f1 = new Fraction(3,4);
new Fraction(4,5);
```


YORK UNIVERSITÉ UNIVERSITY

RQ 4.5

- Predict the outcome of the following statement

```
Fraction f1 = Fraction(3,4);
```


YORK UNIVERSITÉ UNIVERSITY

RQ 4.5

- Which of these are default constructors?
  For such constructors, what would be the initial state of the object?

```
Fraggle x = new Fraggle("hi", 6.5);
Novel y = new Novel("Gone with the Wind");
PainThreshold p = new PainThreshold();
Player p2 = new Player("Charles Oakley", "NBA");
Student s = new Student();
Wozzle w = new Wozzle("");
Wizzle v = new Wizzle(null);
```


YORK UNIVERSITÉ UNIVERSITY

RQ 4.5

- Which of these are default constructors?
  For such constructors, what would be the initial state of the object?

```
Fraggle x = new Fraggle("hi", 6.5);
Novel y = new Novel("Gone with the Wind");
PainThreshold p = new PainThreshold();
Player p2 = new Player("Charles Oakley", "NBA");
Student s = new Student();
Wozzle w = new Wozzle("");
Wizzle v = new Wizzle(null);
```


YORK UNIVERSITÉ UNIVERSITY

RQ

- If an attribute is non-static, can it be `final`?
- Can a method be declared `final`? If so, what's the purpose in doing so?

YORK UNIVERSITÉ UNIVERSITY

RQ 4.34

- If an attribute is `final`, must it also be `static`?

YORK UNIVERSITÉ UNIVERSITY

RQ

- In the following, how many objects are created? How many objects will be garbage collected?

```
Fraction f1 = new Fraction(3,5);
Fraction f2 = new Fraction(3,5);
Fraction f3 = new Fraction(3,5);
Fraction f4 = new Fraction(3,5);
f4 = f1;
f2 = f3;
f3 = null;
```


YORK UNIVERSITÉ UNIVERSITY

RQ 4.23

- Below, the variable `f1` is set to null. Will the object to which it previously referred be deleted? (garbage collected)?

```
Fraction f1 = new Fraction(3,5);
Fraction f2 = new Fraction(3,5);
Fraction f3 = new Fraction(3,5);
f2 = f1;
f1 = null;
f3 = f1;
```


YORK UNIVERSITÉ UNIVERSITY

RQ 4.13, 4.14

- If two objects are deemed equal using `==`, would they also be equal according to the `equals` method?
- If two objects are deemed equal according to the `equals` method, would they also be equal according to `==` ?

YORK UNIVERSITÉ UNIVERSITY

The class `Stock`

- We will use the `Stock` class from type.jar for this example
- A *public* company is a company that offers its stock/ shares for sale to the general public, typically through a stock exchange
- A public company has a full name and is represented by a two-character symbol
  - e.g., name: “Alpha Bravo Co.”, symbol: “.AB”
- At any given point in time, the company’s shares have a **selling price**.
- We use the class `Stock` to encapsulate a single share

YORK UNIVERSITÉ UNIVERSITY

The class `Stock`

- When constructing a `Stock` instance, the client must specify the two-character symbol.
- The `Stock` class' `getName()` accesses the name of the company that corresponds to the stock's two-character stock exchange symbol:

```
ALPHA of BRAVO Company
Alpha of Bravo Company
```

- Whether the name is upper-case or camel-case, this is determined by the boolean flag `titleCaseName`
- The attribute is **public** and **static**

The class `Stock`

- The `Stock` class' `toString()` produces a "nice" string representation consisting of something like:

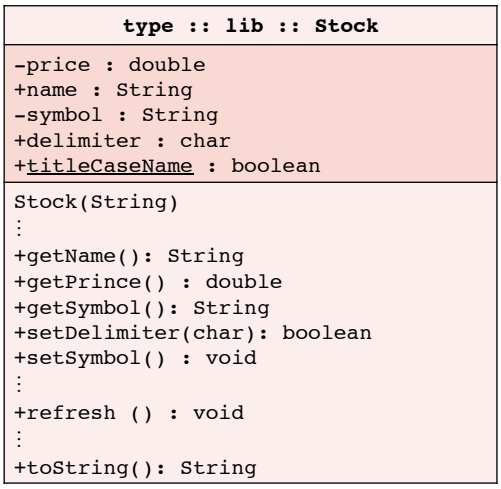
```
.AB*ALPHA of BRAVO Company
.AB:ALPHA of BRAVO Company
.AB+ALPHA of BRAVO Company
.AB ALPHA of BRAVO Company
.AB#ALPHA of BRAVO Company
.AB.ALPHA of BRAVO Company
```

- The character is red is called the delimiter
- The client can specify the character to be used for the delimiter

The class `Stock`

- The `Stock` class' `getPrice()` retrieves the most-recently fetched version of the price. Upon instantiation, the current price is fetched.
- The method `refresh()` will connect to the Stock Exchange server and fetch the current version of the price

UML Diagram



Exercise

- At runtime,
  - how many references will be created?
  - how many objects will be created?
  - do any objects have the same state?
  - predict the output

```
Stock s1 = new Stock(".AB");
Stock s2 = new Stock(".BT");
Stock s3 = new Stock(".XY");
Stock s4 = new Stock(".AB");
output.printf("s1: %s%n", s1.toString());
output.printf("s2: %s%n", s2.toString());
output.printf("s3: %s%n", s3.toString());
output.printf("s1 == s4: %s%n", s1==s4);
output.printf("s1.equals(s4): %s%n", s1.equals(s4));
```


YORK UNIVERSITÉ UNIVERSITY

Exercises 4.11-4.12

- consult the API for class `type.lib.Stock`
  - we will revisit this class in this week's lab exercises

YORK UNIVERSITÉ UNIVERSITY