

Lesson 17: Java Style and Syntax #1 (W05D3)

Balboa High School

Michael Ferraro

September 16, 2015

- PS #2:
 - Turn in your PS #2 paper form.
 - If you need to continue working on PS #2 (online portion), email me ASAP to request that I unlock your problem set. **Note that you won't be eligible for the bonus since we're going over it this period.**
- Create a project called Lesson17 in workspace0 and import these files:
 - `Box.java`
 - `BoxDriver.java`
- Edit `Box.java`, making it look “correct.” For added challenge, use a no-frills text editor like `gedit`!

Students will see how to write a method that takes objects, answer an AP-style FRQ¹, and apply Java formatting guidelines to some poorly formatted code.

¹Free-Response Question

Bonus Round from PS #2

The challenge was to write a method called `makeTransfer()`

Bonus Round from PS #2

The challenge was to write a method called `makeTransfer()`

- The method has to take 2 parameters

Bonus Round from PS #2

The challenge was to write a method called `makeTransfer()`

- The method has to take 2 parameters
 - the amount to be transferred out of `this` account
 - `BankAccount` to which amount would be transferred

Bonus Round from PS #2

The challenge was to write a method called `makeTransfer()`

- The method has to take 2 parameters
 - the amount to be transferred out of this account
 - `BankAccount` to which amount would be transferred
- One of the parameters is an object — a `BankAccount`

Bonus Round from PS #2

The challenge was to write a method called `makeTransfer()`

- The method has to take 2 parameters
 - the amount to be transferred out of this account
 - `BankAccount` to which amount would be transferred
- One of the parameters is an object — a `BankAccount`
 - that's not unusual...we've passed `Strings` to methods, and `Strings` are objects, complete with constructors and methods

The challenge was to write a method called `makeTransfer()`

- The method has to take 2 parameters
 - the amount to be transferred out of this account
 - `BankAccount` to which amount would be transferred
- One of the parameters is an object — a `BankAccount`
 - that's not unusual...we've passed `Strings` to methods, and `Strings` are objects, complete with constructors and methods
 - Since `CheckingAccounts` and `SavingsAccounts` are both subclasses of `BankAccount`, they're both considered `BankAccounts`

Let's build `makeTransfer()`

- First look at a JAR file: Watch as I quickly populate new project with multiple source files!

([teacher link](#))

- Watch the method being built. . .

Distribute the Program!

- A common way to distribute Java apps is as a JAR file

Distribute the Program!

- A common way to distribute Java apps is as a JAR file
- JAR = **J**ava **A**rchive

Distribute the Program!

- A common way to distribute Java apps is as a JAR file
- JAR = **J**ava **A**rchive
- Demo:
 - Using Eclipse to build a *runnable* JAR
 - Running a JAR:
`java -jar JAR_FILE`
 - Examining contents of a JAR:
`jar tf JAR_FILE`

AP-Style Thought Questions

Consider the method `makeTransfer()` that has been added to the `BankAccount` class.

1. What would be the behavior of `makeTransfer()` if a *negative* amount were passed to it?

AP-Style Thought Questions

Consider the method `makeTransfer()` that has been added to the `BankAccount` class.

2. If `acctA` has a balance of \$80 and `acctB` has a balance of \$120, what would be state of the two accounts' balances if the line below is evaluated?

```
acctA.makeTransfer(acctB, 90);
```

AP-Style Thought Questions

Consider the method `makeTransfer()` that has been added to the `BankAccount` class.

3. What would happen in this case?

```
acctB.makeTransfer(acctB, 20);
```

- For this topic area — Java style and syntax — you will read about the various style rules and learn them on your own (with a little help from me).
- Read §§3.1-3.5 in the [textbook](#).²
- **As you read**, answer questions in §2.2 of PS #3
- Once finished, move on to the next slide.

²credentials provided in class!

- Read more! Litvin §3.6 (<1pg)
- Work on Litvin §3.7, “Lab: Correcting Syntax Errors”
The code for `MovingDisk.java` is available [here](#).
- Continue working on on PS #3
- Next class: Javadoc demonstration

Finish §§1–2 of PS #3.