Chapter 4

Defining your own classes

CS180 Recitation, 03 February 2006

Department of Computer Science

Purdue University

Reminders

- Assignment 2 was due last night.
- Visit Academic Integrity website.
- Use lab section when doing turnin for projects.
- Assignment 3 available

  due Thursday, February 16 – 2 weeks

- Exam 1 – Thurs, Feb. 9, 7-8pm, LILY G126

Reminders (cont'd)

- Project 1 re-grades:

  If you received less than 50 points on the first project, you may resubmit no later than Friday, February 10, 10:30pm for a re-grade. A re-grade can receive a maximum of 50 points.

- From now on:

  Projects which don't compile receive ZERO pts.

  Re-grades only allowed for first project.

Problem Statement

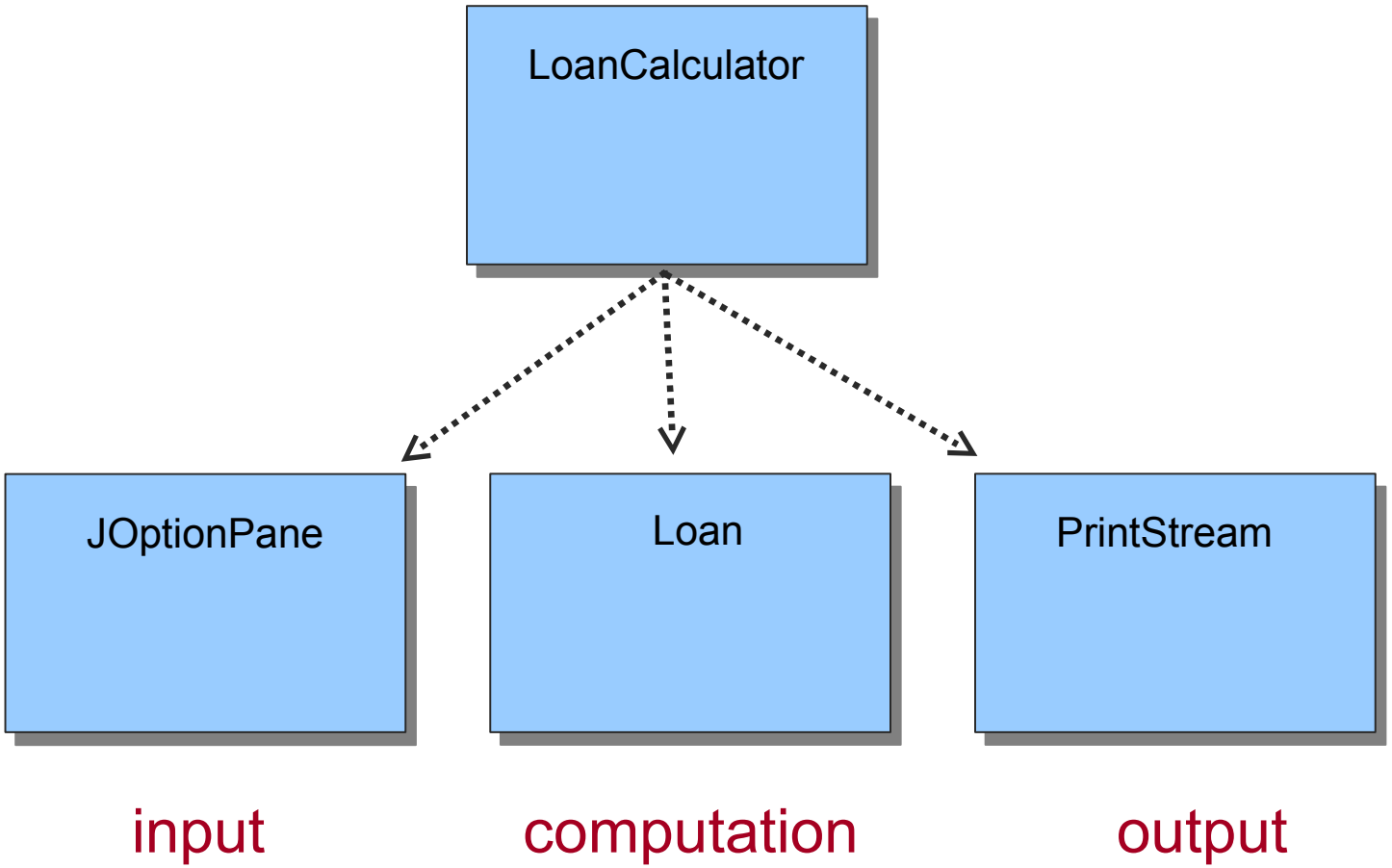
- Problem statement:

  Write a loan calculator program that computes both monthly and total payments for a given loan amount, annual interest rate, and loan period.

Overall Plan

- Tasks:
  - Get three input values: **loanAmount**, **interestRate**, and **loanPeriod**.
  - Compute the monthly and total payments.
  - Output the results.

Required Classes



Development Steps

- We will develop this program in five steps:

1. Start with the main class LoanCalculator. Define a temporary placeholder Loan class.
2. Implement the input routine to accept three input values.
3. Implement the output routine to display the results.
4. Implement the computation routine to compute the monthly and total payments.
5. Finalize the program.

Step 1 Design

- The methods of the LoanCalculator class

Method	Visibility	Purpose
start	public	Starts the loan calculation. Calls other methods
computePayment	private	Give three parameters, compute the monthly and total payments
describeProgram	private	Displays a short description of a program
displayOutput	private	Displays the output
getInput	private	Gets three input values

Step 1 Code

Directory: Chapter4/Step1

Source Files:

LoanCalculator.java
Loan.java

Step 1 Test

- In the testing phase, we run the program multiple times and verify that we get the following output

```
inside describeProgram
inside getInput
inside computePayment
inside displayOutput
```

Step 2 Design

- Design the input routines
  - LoanCalculator will handle the user interaction of prompting and getting three input values
  - LoanCalculator calls the setAmount, setRate and setPeriod of a Loan object.

Step 2 Code

Directory: Chapter4/Step2

Source Files:

LoanCalculator.java
Loan.java

Step 2 Test

- We run the program numerous times with different input values
- Check the correctness of input values by echo printing

```
System.out.println("Loan Amount: $"
                           + loan.getAmount());

System.out.println("Annual Interest Rate:"
                           + loan.getRate() + "%");

System.out.println("Loan Period (years):"
                           + loan.getPeriod());
```

Step 3 Design

- We will implement the displayOutput method.
- We will reuse the same design we adopted in Chapter 3 sample development.

Only the computed values (and their labels) are shown

```
Monthly payment:          $ 143.47
Total payment:            $ 17216.50
```

Both the input and computed values (and their labels) are shown.

```
For
Loan Amount:              $ 10000.00
Annual Interest Rate:       12.0%
Loan Period (years):        10

Monthly payment is        $ 143.47
  TOTAL payment is        $ 17216.50
```

Step 3 Code

Directory: Chapter4/Step3

Source Files:

LoanCalculator.java
Loan.java

Step 3 Test

- We run the program numerous times with different input values and check the output display format.
- Adjust the formatting as appropriate

Step 4 Design

- Two methods getMonthlyPayment and getTotalPayment are defined for the Loan class
- We will implement them so that they work independent of each other.
- It is considered a poor design if the clients must call getMonthlyPayment before calling getTotalPayment.

Step 4 Code

Directory: Chapter4/Step4

Source Files:

LoanCalculator.java
Loan.java

Step 4 Test

- We run the program numerous times with different types of input values and check the results.

Input			Output (shown up to three decimal places only)	
Loan Amount	Annual Interest Rate	Loan Period (in Years)	Monthly Payment	Total Payment
10000	10	10	132.151	15858.088
15000	7	15	134.824	24268.363
10000	12	10	143.471	17216.514
0	10	5	0.000	0.000
30	8.5	50	0.216	129.373

Step 5: Finalize

- We implement the **describeProgram** method.
- We will format the monthly and total payments to two decimal places using DecimalFormat.

Directory: Chapter4/Step5

Source Files (final version):

LoanCalculator.java
Loan.java

Questions / Comments ?