Building Java Programs

Chapter 6 Lecture 15: Line-Based File Input

reading: 6.3 - 6.5

(Slides adapted from Stuart Reges, Hélène Martin, and Marty Stepp)



Cities problem

• Consider the following census data:

- City
 State
 Population
 Latitude
 Longitude

 Seattle
 WA
 616627
 47621800
 -122350326

 New_York
 NY
 8391881
 40669800
 -73943849

 Los_Angeles
 CA
 3831868
 34112101
 -118411201
- Write a program that plots any cities bigger than a given city:

Plot all cities as large as which city? **<u>seattle</u>**

Cities problem (output)

Cities with at least 616627 people

.

Cities problem (output)

- If the city is not found
 - print

"<city_name>" not found

where <city_name> is exactly what the user typed in

Also we should not create a DrawingPanel

Cities problem (continued)

Plan of attack

- Get city name from user
- Find the city in the file
 - Find the population of the city
- Scan the whole file plotting each city with greater population
 - Keep track of how many cities are plotted
- Draw the border and stats (as Strings)

"Chaining"

- main should be a concise summary of your program.
 - It is bad if each method calls the next without ever returning (we call this *chaining*):



- A better structure has main make most of the calls.
 - Methods must return values to main to be passed on later.



Cities Structure

 A poor structure would have each method call the next method, passing the required values on



 A better structure would have main make calls to each method, delegating tasks

