



## Project 14: Comma Separated Dogs

---



## Project 14: Comma Separated Dogs

---

- This is a variation of the Dogs from File program in Project 13.
- Write a constructor for class Dog that takes a single line of a Comma Separated Values file as its only parameter.
  - Initializes the Dog object with information from the CSV line.
- Let the new constructor have the signature
  - `Dog(String dog_info)`



## Project 14: Comma Separated Dogs

---

- You may use the posted solution for Project 13 as a starting point for this project, if you wish.
- It is OK to copy code from any file posted on the class web site.



# Assignment

---

- Add a serialization method to the Dog class.
  - Returns a single String consisting of all member variables, separated by commas, if object is valid.
  - Returns a blank line (empty string) if object is invalid.



# Assignment

---

- The constructor should verify that the number of comma separated items in the input line is correct.
  - Set private boolean member variable isValid to false if the input line does not have the right number of items.
  - Class should include an accessor method for isValid.
- The Dog class should include a toString method that returns a string with the dog's information in a printable format (single line).
  - Provides an appropriate result for an invalid object.



# Assignment

---

Create a test driver called DogsFromCSV.java

- Accepts input from the keyboard for a file name.
- Creates a Scanner object for the file.
- Reads each line from the file and invokes the new Dog constructor, passing it the CSV line.
- Outputs each Dog's information using println.
- Adds 1 to the dog's age and writes the updated information to a new CSV file having the input file name with "2" appended.
  - Example: dogs.txt > dogs2.txt



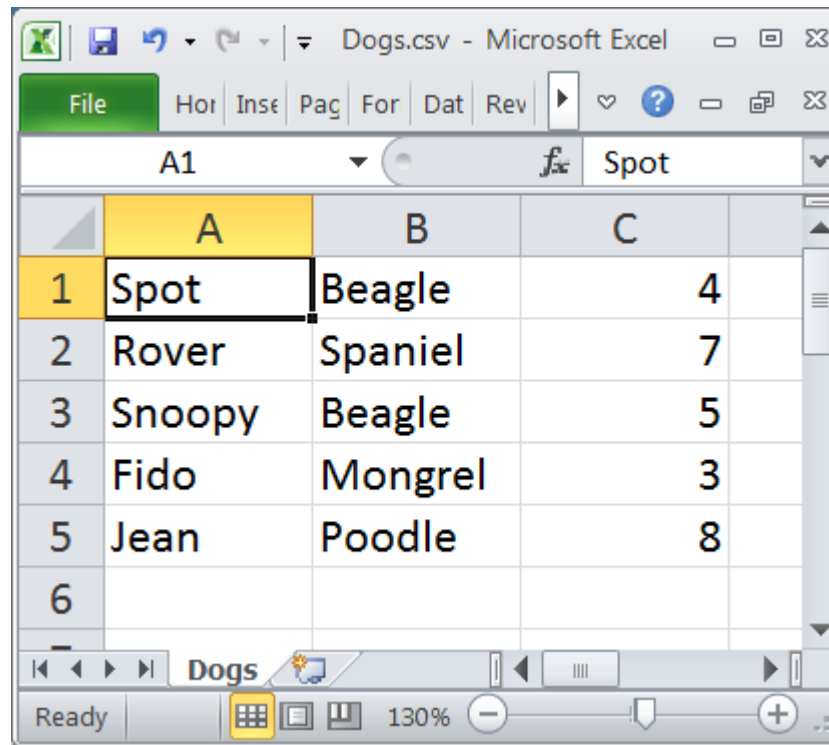
# DogsFromFile.java

---

- If the file specified by the user does not exist, output an error message and let the user try again.

# Test File

- You can download a test file from the class web site:
- [http://www.csee.usf.edu/~turnerr/Programming\\_Concepts/Downloads/Project\\_14\\_CSV\\_Dogs/](http://www.csee.usf.edu/~turnerr/Programming_Concepts/Downloads/Project_14_CSV_Dogs/File_Dogs.csv) File Dogs.csv



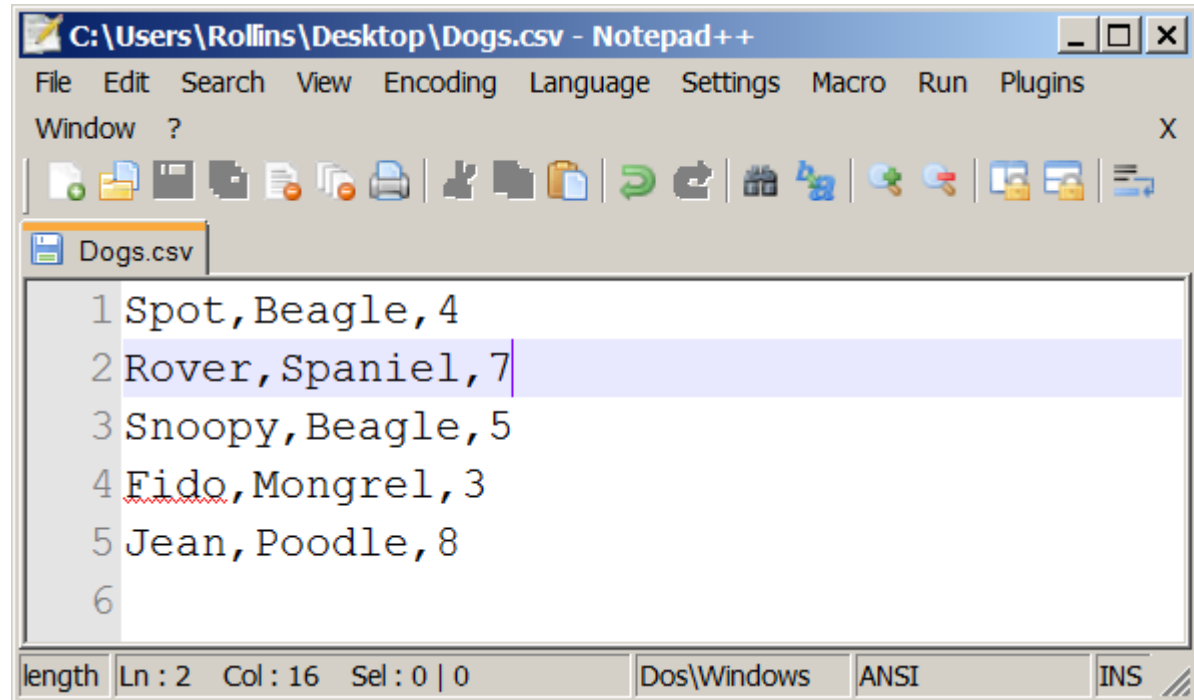
The screenshot shows a Microsoft Excel window titled 'Dogs.csv - Microsoft Excel'. The spreadsheet contains a table with 6 rows and 4 columns. The first row is the header, with columns labeled A, B, and C. The data rows are as follows:

	A	B	C
1	Spot	Beagle	4
2	Rover	Spaniel	7
3	Snoopy	Beagle	5
4	Fido	Mongrel	3
5	Jean	Poodle	8
6			

The status bar at the bottom indicates 'Ready' and '130%' zoom.



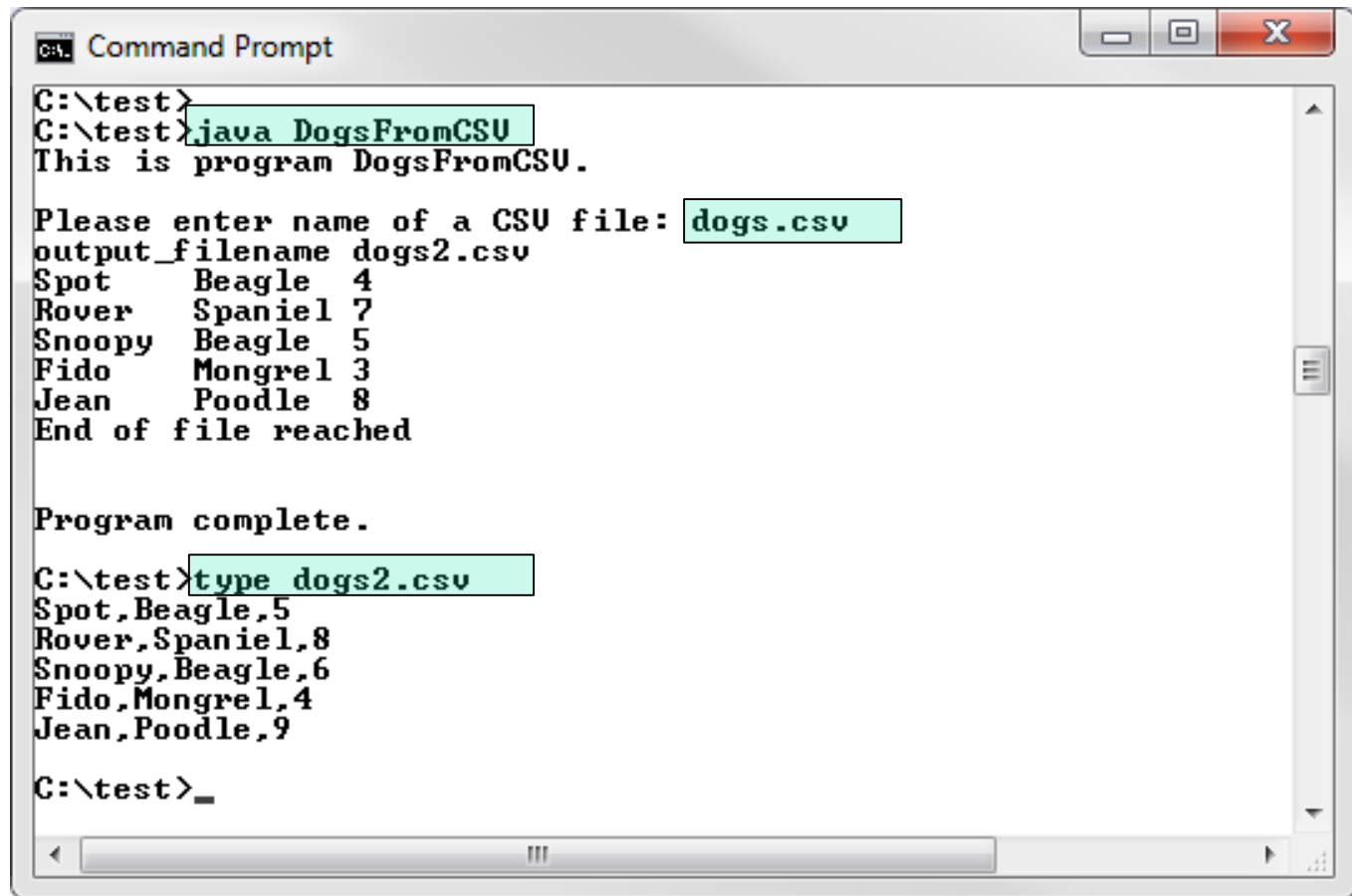
# Dogs.csv in Notepad++



A screenshot of the Notepad++ application window. The title bar reads "C:\Users\Rollins\Desktop\Dogs.csv - Notepad++". The menu bar includes File, Edit, Search, View, Encoding, Language, Settings, Macro, Run, and Plugins. The toolbar contains various icons for file operations and editing. The text area shows a CSV file with five rows of data, each preceded by a line number. The second row, "2 Rover, Spaniel, 7", is selected. The status bar at the bottom shows "length Ln : 2 Col : 16 Sel : 0 | 0", "Dos\Windows", "ANSI", and "INS".

Line	Name	Breed	Age
1	Spot	Beagle	4
2	Rover	Spaniel	7
3	Snoopy	Beagle	5
4	Fido	Mongrel	3
5	Jean	Poodle	8
6			

# Test Run with Good Dogs



```
CA: Command Prompt
C:\test>
C:\test>java DogsFromCSU
This is program DogsFromCSU.

Please enter name of a CSU file: dogs.csv
output_filename dogs2.csv
Spot    Beagle  4
Rover   Spaniel 7
Snoopy  Beagle  5
Fido    Mongrel  3
Jean    Poodle   8
End of file reached

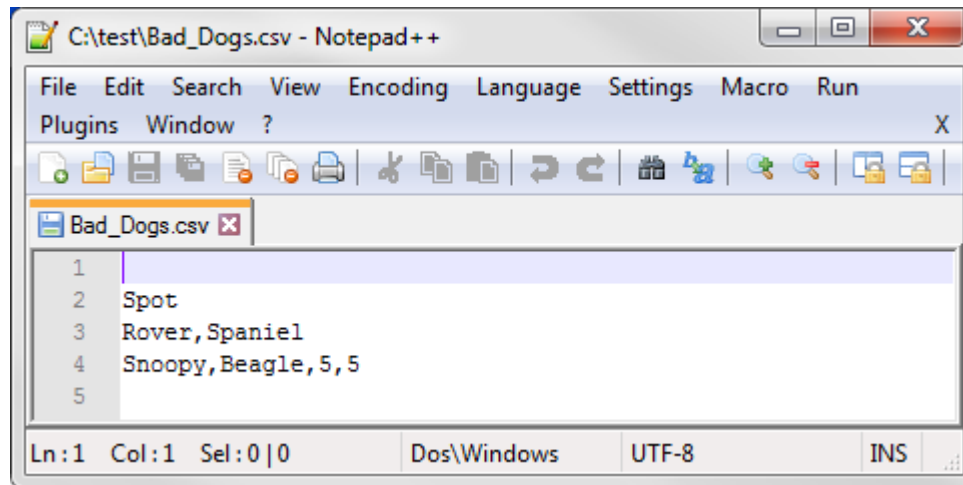
Program complete.

C:\test>type dogs2.csv
Spot,Beagle,5
Rover,Spaniel,8
Snoopy,Beagle,6
Fido,Mongrel,4
Jean,Poodle,9

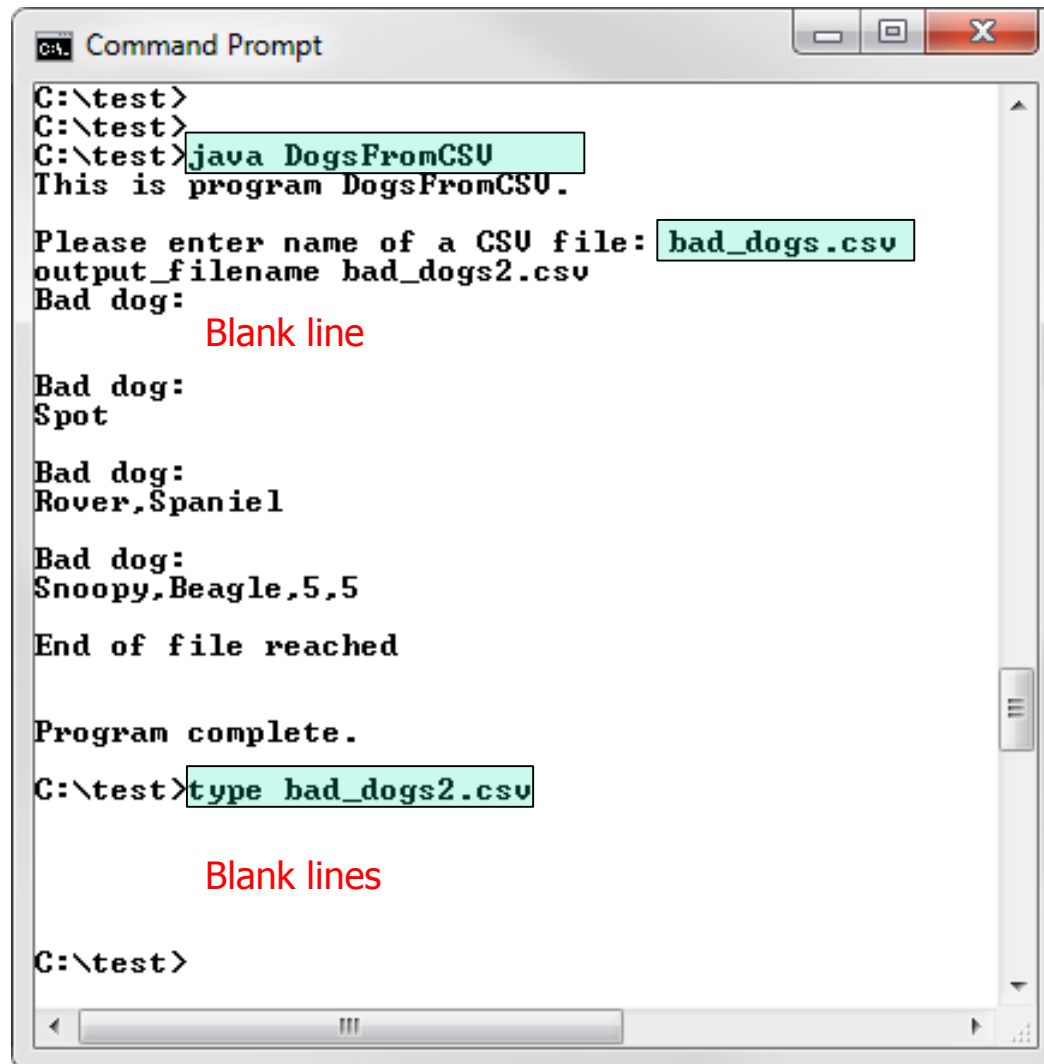
C:\test>
```

# Bad Dogs

- The Downloads area also has a file of invalid lines:
  - Bad\_Dogs.csv
- You can use this file to test your error processing.



# Test Run with Bad Dogs



```
C:\> Command Prompt
C:\test>
C:\test>
C:\test> java DogsFromCSV
This is program DogsFromCSV.

Please enter name of a CSV file: bad_dogs.csv
output_filename bad_dogs2.csv
Bad dog:

Blank line

Bad dog:
Spot

Bad dog:
Rover,Spaniel

Bad dog:
Snoopy,Beagle,5,5

End of file reached

Program complete.
C:\test> type bad_dogs2.csv

Blank lines

C:\test>
```



# Implementation Tips

---

- Use `Integer.parseInt()` to convert the string representation of an integer to an integer.
  - Example:

```
int i = Integer.parseInt("1234");
```

sets the variable `i` to the integer value 1234.
- To create the output file name you can write

```
String outfile_name =  
    infile_name.replace(".csv", "2.csv");
```



# Submission

---

- Put your Java source files into a folder and zip it.
- Submit your zipped folder via Canvas Assignments.
- Project is due by 11:59 PM
  - Sunday, April 17 **All Sections**
- Recommendation:

Do this project in your lab session or help session.



# Ground Rules

---

- It is OK to *discuss* the project with other students BUT
  - Do not share your code with other students.
  - Before or after submitting the project.
- Do not copy any other student's code.
  - Or even look at it.
- Do not let anyone copy or examine your code.



# Ground Rules

---

Except for code posted on the class web site

- Do not copy code from the Internet
  - or any other source (other than the textbook.)
- Do not ask for help on an Internet forum.
  - If you need help, ask your instructor or a TA.
  - Come to lab and help sessions.
- Write your own code.