Java Review Session 4

CS 5004

Topics Covered

Java Syntax

- Generics
- File I/O
- Converting Strings to numbers
- Format Flags
- Format Types
- Command line arguments

Generics

- Generic programming - is the creation of programming constructs that can be used with many different types
- We already saw that the ArrayList<> class uses the technique of generic programming (this way you can collect elements of different types)
- A generic class - has one or more type parameters (used for specifying the type of its elements)
- You supply a variable for each type parameter
- ArrayList<E> - E is the type variable that denotes the element type
  - Also used whenever you need to denote this return type for a method declaration
  - public void add(E element) and public E get(int index)
- It is customary to use short, uppercase names for the variables

Generics continued

- In order to use a generic class, you need to instantiate the type parameter (supply the actual type - any class or interface type - not primitives!)
- When you instantiate a generic class, the type that you supply replaces all instances of that type variable in the class you declared
- Why do we use generics?
  - Type safety
  - More easily readable code

Creating a generic class

Type parameter naming conventions

Type Variable	Meaning
`E`	Element type in a collection
`K`	Key type in a map
`V`	Value type in a map
`T`	General type
`S, U`	Additional general types

Syntax 18.1 Declaring a Generic Class

Syntax

```
modifier class GenericClassName<TypeVariable1, TypeVariable2, . . .>
{
   instance variables
   constructors
   methods
}
```

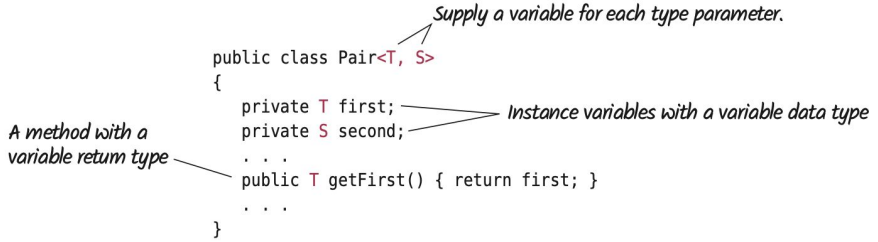

Image: Cay Horstmann

File I/O - (reading and writing text files)

- There are multiples ways to read in a file in Java
  - Can use: bufferedreader, FileInputStream, Scanner, etc.
- We will take a look at reading in files using a scanner
- You can write output by creating a PrintWriter object
- File I/O demo

Format flags and types - format specifier structure

First character %, then optional flags that modify the format (table 2), then the field width (the total number of characters in the field followed by an optional precision for floating point numbers), and ends with the format type (table 3)

Table 2 Format Flags

Flag	Meaning	Example
-	Left alignment	1.23 followed by spaces
0	Show leading zeroes	001.23
+	Show a plus sign for positive numbers	+1.23
(	Enclose negative numbers in parentheses	(1.23)
,	Show decimal separators	12,300
^	Convert letters to uppercase	1.23E+1

Table 3 Format Types

Code	Type	Example
d	Decimal integer	123
f	Fixed floating-point	12.30
e	Exponential floating-point	1.23e+1
g	General floating-point (exponential notation is used for very large or very small values)	12.3
s	String	Tax:

Image: Cay Horstmann

Command line arguments

- When you type the name of the program to run in the command line this is called “invoking the program from the command line”
  - You type the name of the program, but can also type additional information that the program can use (this Strings are called command line arguments)
- `java myProgram input.txt`
- This program receives input.txt as a command line argument
  - It is entirely up to the program what to do with these Strings (arguments)
  - When to use this? Consider GUI for not frequent users and command line for frequent users
- Your program receives its command line arguments in the args parameter of the main method
  - args[0]: input.txt