Sukkur IBA

Dr. Faheem Akhtar Rajput

Object Oriented Programming (JAVA)

Lecture 7 & 8

Fundamental Elements of Java

- There are 3 fundamental elements of JAVA:
  - Data Types
  - Variables
  - Arrays

Primitive Types

- 8 Primitive types of data, divided in four different categories.
  - **Integer:** `byte`, `short`, `int` **and** `long`
  - **Floating-Point:** `float`, `double`
  - **Character:** `char`
  - **Boolean:** `boolean`

Integers

- Java defines four integer types: `byte`, `short`, `int`, and `long`.

Type		Size
byte	-	1
short	-	2
int	-	3
long	-	4

- All are signed.

Type	Storage Requirement	Range (Inclusive)
int	4 bytes	–2,147,483,648 to 2,147,483, 647 (just over 2 billion)
short	2 bytes	–32,768 to 32,767
long	8 bytes	–9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
byte	1 byte	–128 to 127

Example - 1

```
// Compute distance light travels using long variables.
class Light {
        public static void main(String args[]) {
                int lightspeed;     long days;     long seconds;   long distance;
                // approximate speed of light in miles per second
                lightspeed = 186000;
                days = 1000; // specify number of days here
                seconds = days * 24 * 60 * 60; // convert to seconds
                distance = lightspeed * seconds; // compute distance
                System.out.print("In " + days);
                System.out.print(" days light will travel about ");
                System.out.println(distance + " miles.");
        }
}
```

Floating-Point

- 2 floating point or real number data types: `float` and `double`

Type		size		precision
float	-	4	-	8
double	-	8	-	16

Type	Storage Requirement	Range
float	4 bytes	approximately ±3.40282347E+38F (6–7 significant decimal digits)
double	8 bytes	approximately ±1.79769313486231570E+308 (15 significant decimal digits)

Example - 2

```
// Compute the area of a circle.
class Area {
        public static void main(String args[]) {
                double pi, r, a;
                r = 10.8;               // radius of circle
                pi = 3.1416;            // pi, approximately
                a = pi * r * r;         // compute area
                System.out.println("Area of circle is " + a);
        }
}
```

Character

- It includes only one type: `char`

Type	Size	Value
char	16	‘A’

- Java is able to use **18** international languages
- UNICODE character set
- ASCII character set

Example - 3

```
// Demonstrate char data type.
class CharDemo {
	public static void main(String args[]) {
		char ch1, ch2;
		ch1 = 88; // code for X
		ch2 = 'Y';
		System.out.print("ch1 and ch2: ");
		System.out.println(ch1 + " " + ch2);
	}
}
```

Boolean

- Boolean is either `true` or `false`

Type	Size	Value
boolean	0	`true/false`

- Flip-flop: 1 bit of memory is the size of boolean

Integer Literals

- Any whole number is a literal (1, 2 15, 42..)
- Two other bases can also be used in integer literals
  - Octal (8 base) - leading (0) zero - 0 to 7
  - Hexadecimal (16 base) - leading (0x) zero-x 0 to 15
- No error when assigned values to `short`, `byte` or even `char`
- Will generate an error while assigning `long` value

Float boolean and char type Literals

- Floating Point
  - Within the range literals can be positive or negative
  - Default precision is double (put *f* for float)
- Boolean
  - Only two values true or false
  - `true` doesn't equal to 1 and `false` doesn't equal to 0
- Character
  - Unicode character set.
  - There is also a mechanism for directly entering the value of a character in octal or hexadecimal. In Octal - '\141' for 'a'. In Hexadecimal (\u). '\ua432' for Japanese letter.
  - Several escape sequences allowed '\''

String Literal

- Strings are objects in java, while in other languages they are implemented as array of characters.
- With extensive string handling capabilities are powerful and easy
- Must remain on same line, no line continuation for string.

```
"Hello World"

"two\nlines"

 "\"This is in quotes\""
```

Escape Sequence

Escape Sequence	Description
\ddd	Octal character (ddd)
\uxxxx	Hexadecimal Unicode character (xxxx)
\'	Single quote
\"	Double quote
\\	Backslash
\r	Carriage return
\n	New line (also known as line feed)
\f	Form feed
\t	Tab
\b	Backspace

Variables

- Declaring a variable: all variables must be declared before they can be used.

type identifier [= value][, identifier [= value] ...] ;

- Dynamic Initialization: Java allows variables to be initialized dynamically

Questions?

Sukkur IBA

Thanks…