Datatypes and Wrapper Class

Colorado State University
Department of Computer Science

Slides Originally Created by Albert Lionelle (Albert.Lionelle@colostate.edu),
updated by Marcia Moraes (marcia.moraes@colostate.edu)

Announcements

TODO Reminders:
Readings are due **before** lecture

- Reading 8 (zybooks) – you should have already done that ☺
- Lab 05
- Reading 9 (zyBooks) – you should have already done that ☺
- Lab 06
- Reading 10 (zybooks)
- Keep practicing your RPAs in a spaced and mixed manner ☺

https://images.squarespace-cdn.com/content/v1/5e355fb2c950981b57643f21/1580903818935-B9JO2ZTBHH5N4IKXC0WF/Lettering_Practice_Nov2019_lowres.jpg?format=1000w

Recall Activity - Attendance

What is a wrapper class?
Explain using your own words and providing examples.

Binary

Binary – two state system

- 1 for on
- 0 for off

Bit

- Each 0 and 1 is called a bit
- 8 bits is called a byte
- Contains 255 states (128+64+32+16+8+4+2+1)

Binary

- Every bit is a exponential of 2
  - 1 + 2 + 4 + 8, etc

Take

$$1001$$

$$= 2^3 * 1 + 2^2 * 0 + 2^1 * 0 + 2^0 * 1$$

$$= 8 + 0 + 0 + 1$$

$$= 9$$

Binary

```
public class BinaryToDecimal {
    public static int convertBinaryToDecimal(String binaryString){
        int dec = Integer.parseInt(binaryString, 2);
        return dec;
    }
    public static void main(String args[]){
        int answer = convertBinaryToDecimal("110");
        System.out.println(answer);
        System.out.println(Integer.toBinaryString(answer));
        answer = convertBinaryToDecimal("1001");
        System.out.println(answer);
        System.out.println(Integer.toBinaryString(answer));
    }
}
```

Numeric data types

Declaration	Size	Supported number range
byte myVar;	8 bits	-128 to 127
short myVar;	16 bits	-32,768 to 32,767
int myVar;	32 bits	-2,147,483,648 to 2,147,483,647
long myVar;	64 bits	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807

Declaration	Size	Supported number range
float x;	32 bits	-3.4×10^{38} to 3.4×10^{38}
double x;	64 bits	-1.7×10^{308} to 1.7×10^{308}

Casting:

This is also why an int will automatically convert to a double, but a double can't convert to an int without explict casting - possible loss of information!

Numeric data types – Formatting to Hexadecimal

```
public class FormatHexadecimal {
    public static void main(String args[]){
        short red = 127;
        short green = 135;
        short blue = 255;
        String htmlCode = String.format("#%02X%02X%02X", red, green, blue);
        System.out.println(htmlCode);
    }
}
```

Can we use byte instead of short?

#7F87FF

Wrapper Class

- A *primitive type* variable directly stores the data for that variable type, such as int, double, or char.
- A *reference type* variable can refer to an instance of a class, also known as an object.
- *Wrapper classes* that are built-in reference types that augment the primitive types

Wrapper Class

Reference type	Associated primitive type
Character	char
Integer	int
Double	double
Boolean	boolean
Long	long

Wrapper Class Conventions

- Autoboxing - automatic conversion of primitive types to the corresponding wrapper classes

Scenario	Examples

Assign primitive type to wrapper class variable	`Double floorArea = 20.25; // Autoboxing of 20.25 to a Double`
	`Integer calcResult;`
	
	`calcResult = 5 / 2; // Autoboxing of expression result to Integer`
	
	`int num1 = 23;`
	`Integer num2 = num1; // Autoboxing of num1 to Integer`

Pass primitive type to a method with wrapper class parameter	`public void setRate(Double rate) {`
	`  // ...`
	`}`
	
	`setRate(50.2); // Autoboxing of 50.2 to Double`
	
	`double newRate = 97.2;`
	`setRate(newRate); // Autoboxing of newRate to Double`

Wrapper Class Conventions

- Unboxing - automatic conversion of wrapper class objects to the corresponding primitive types

Scenario	Examples

Assign wrapper class variable to primitive type	`Double num1 = 3.14;`
	`Character letter1 = 'A';`
	
	`double num2 = num1; // Unboxing of Double to double`
	`char letter2 = letter1; // Unboxing of Character to char`

Pass wrapper class variable to a method with primitive type parameter	`public void setInitial(char letter) {`
	`  // ...`
	`}`
	
	`Character userInitial = 'Z';`
	`setInitial(userInitial); // Unboxing of userInitial to a char`

Combine wrapper class and primitive types in expression	`Double currTemp = 95.2;`
	`double tempDiff = 100.0 - currTemp; // Unboxing of currTemp to double`
	
	`Integer numItems = 11;`
	
	`if (numItems % 2 == 0) { // Unboxing of numItems to int`
	`  // ...`
	`}`

Character Wrapper Class

- char is a primitive
  - No methods by itself
- Character "wrapper" exists
  - Methods (mostly static) to help you learn about **char**
- Common and useful methods
  - Character.isDigit(char)
    - Example:

```
char chDigit = '9';

boolean dig = Character.isDigit(chDigit); // true
```

  - Character.isWhitespace(char)
    - All whitespace including \t and \n
  - Character.isLetter(char)
- These are often paired with String charAt, and loops!

Coding Along

- Write a method in Java that receives a string as a parameter and removes the whitespaces from the string and print the number of whitespaces removed.

Solution to the Code Along


```
String str = “hello  how are  you?”;
int counter =  0;
String nospace = “”;
for(int i = 0; i < str.length(); i++) {
   char tmp = str.charAt(i);
   if(Character.isWhitespace(tmp)) {
     counter++;
   }else {
      nospace += tmp;
   }
}
System.out.println(nospace);
System.out.printf(“Whitespace removed %d%n”,
counter)
```

```
hellohowareyou
Whitespace removed 5
```

Do the In Class Activity.