

JAVA

기본 프로그래밍 03

Objective

Java Coding Convention

Character Encodings

Creating Object

Calling a Method

Java Library

- ▶ Calendar
- ▶ Math
- ▶ TimeZone
- ▶ String
- ▶ Numeric Classes

Java Coding Convention

Google Java Style Guide

Google Java Style Guide

Table of Contents

[1 Introduction](#)

- [1.1 Terminology notes](#)
- [1.2 Guide notes](#)

[2 Source file basics](#)

- [2.1 File name](#)
- [2.2 File encoding: UTF-8](#)
- [2.3 Special characters](#)

[3 Source file structure](#)

- [3.1 License or copyright information, if present](#)
- [3.2 Package statement](#)
- [3.3 Import statements](#)
- [3.4 Class declaration](#)

[4 Formatting](#)

- [4.1 Braces](#)
- [4.2 Block indentation: +2 spaces](#)
- [4.3 One statement per line](#)
- [4.4 Column limit: 100](#)
- [4.5 Line-wrapping](#)

[4.6 Whitespace](#)

- [4.7 Grouping parentheses: recommended](#)
- [4.8 Specific constructs](#)

[5 Naming](#)

- [5.1 Rules common to all identifiers](#)
- [5.2 Rules by identifier type](#)
- [5.3 Camel case: defined](#)

[6 Programming Practices](#)

- [6.1 @Override: always used](#)
- [6.2 Caught exceptions: not ignored](#)
- [6.3 Static members: qualified using class](#)
- [6.4 Finalizers: not used](#)

[7 Javadoc](#)

- [7.1 Formatting](#)
- [7.2 The summary fragment](#)
- [7.3 Where Javadoc is used](#)

Character Encodings

ASCII and UNICODE

	ASCII	UNICODE
Comparison Between ASCII and UNICODE	A character encoding standard for electronic communication	A computing industry standard for consistent encoding, representation, and handling of text expressed in most of the world’s writing systems
	Stands for American Standard Code for Information Interchange	Stands for Universal Character Set
	Supports 128 characters	Supports a wide range of characters
	Uses 7 bits to represent a character	Uses 8bit, 16bit or 32bit depending on the encoding type
	Requires less space	Requires more space

Source : <https://medium.com/@vanvlymenpaws/ascii-vs-unicode-4174def5c09d#:~:text=ASCII%20has%20its%20equivalent%20in,English%2C%20Arabic%2C%20Greek%20etc.>

Q1* : Find out “UTF-8” encoding

P1

Converting Character to ASCII Code

- ▶ Print ASCII code converted numbers and alphabets
- ▶ Hint, the sample code below

```
16 public static void main(String[] args) {  
17     byte a1 = 'a';  
18     byte a2 = 'A';  
19     System.out.println(a1);  
20     System.out.println(a2);  
21 }
```

Problems @ Javadoc Declaration Console

<terminated> Main [Java Application] C:\Users\CTC#.p2#poo

97
65

Converting Characters to Opposite Case

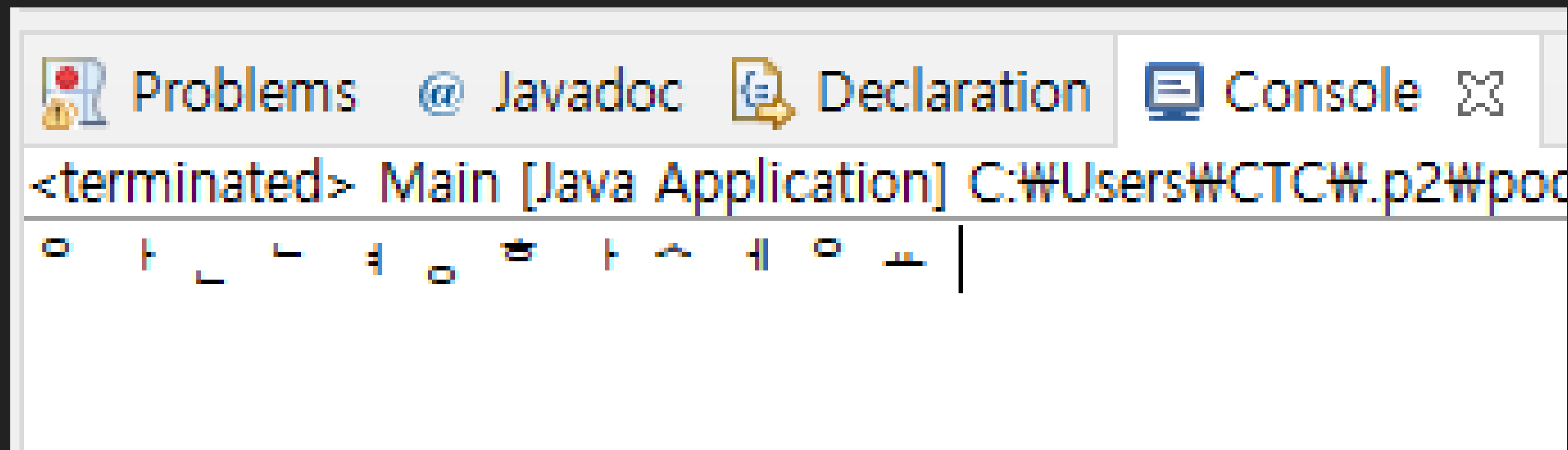
- ▶ User inputs a string
- ▶ Print the converted characters

I am a boy
i AM A BOY

P3

Split Korean Characters into “First”, “Neutral” and “Final” Characters

- ▶ User inputs a string in Korean
- ▶ Print the split characters
- ▶ Hint, Unicode and Eclipse configuration



Creating Object

General Way to Create an Object

- ▶ Using new keyword is the most common and general way to create an object
- ▶ `ClassName objectName = new ClassName();`

ex)

```
Computer computer = new Computer();
```

```
Internet internet = new Internet();
```

```
Phone phone = new Phone();
```


Calling a Method

Calling a Method from the Object

► ObjectName.MethodName()

```
17 public class Main extends Character {  
18     public static void main(String[] args) {  
19         // TODO Auto-generated method stub  
20         Date date = new Date();  
21         System.out.println(date.toString());  
22         System.out.println(date.getTime());  
23     }  
24 }
```

Problems @ Javadoc Declaration Console

<terminated> Main [Java Application] C:\Users\kopo\p2

Sat Mar 06 01:12:00 KST 2021

1614960720134

Calendar Class(Cont'd)

Calendar

- ▶ An abstract class that provides methods for converting date between a specific instant in time and a set of calendar fields such as MONTH, YEAR, HOUR, etc.

```
11 public static void main(String[] args) {  
12     // TODO Auto-generated method stub  
13     Calendar calendar = Calendar.getInstance();  
14     System.out.println(calendar.getTime());  
15     System.out.println(calendar.get(Calendar.YEAR));  
16     System.out.println(calendar.get(Calendar.MONTH));  
17     System.out.println(calendar.get(Calendar.DATE));  
18     System.out.println(calendar.get(Calendar.DAY_OF_WEEK));  
19     System.out.println(calendar.get(Calendar.DAY_OF_MONTH));  
20     System.out.println(calendar.get(Calendar.DAY_OF_YEAR));  
21     System.out.println(calendar.get(Calendar.HOUR));  
22     System.out.println(calendar.get(Calendar.HOUR_OF_DAY));  
23     System.out.println(calendar.get(Calendar.MINUTE));  
24     System.out.println(calendar.get(Calendar.SECOND));  
25     System.out.println(calendar.get(Calendar.MILLISECOND));  
26 }
```

Problems @ Javadoc Declaration Console

<terminated> Main [Java Application] C:\Users\#kopo#\p2\#p

Thu Feb 11 11:25:21 KST 2021

2021

1

11

5

11

42

11

11

25

21

291

P4 : Find out the range of the return values

Calendar Class

Date and Time Calculation

```
11 public static void main(String[] args) {  
12     // TODO Auto-generated method stub  
13     Calendar calendar = Calendar.getInstance();  
14     System.out.println(calendar.getTime());  
15     System.out.println(calendar.get(Calendar.YEAR));  
16     System.out.println(calendar.get(Calendar.YEAR) + 1);  
17     System.out.println(calendar.get(Calendar.MONTH));  
18     System.out.println(calendar.get(Calendar.MONTH) + 3);  
19     System.out.println(calendar.get(Calendar.DATE));  
20     System.out.println(calendar.get(Calendar.DATE) - 7);  
21 }
```

Problems Javadoc Declaration Console

<terminated> Main [Java Application] C:\Users\kopo\p2\p

Thu Feb 11 11:32:52 KST 2021
2021
2022
1
4
11
4

P5 : Try this with the other parameters such as Calendar.HOUR

Calendar Application

- ▶ User inputs the two numbers for year and month
- ▶ Print the calendar

2021
2

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28						

Calendar Application II

- ▶ User inputs the two dates
- ▶ Print the days between the first and the second dates
(Calculate Duration)

20200203

20200212

8

Calendar Application III

- ▶ User inputs a date and a number for adding days
- ▶ Print the new date after adding the input day to the start date

20200203

8

20200212

Math Class

Math Class

- Perform mathematical tasks on numbers

```
11 public static void main(String[] args) {  
12     // TODO Auto-generated method stub  
13     System.out.println(Math.min(3, 13));  
14     System.out.println(Math.max(123, 54));  
15     System.out.println(Math.abs(-23.1));  
16     System.out.println(Math.sqrt(3));  
17     System.out.println(Math.random());  
18 }
```

Problems @ Javadoc Declaration Console

<terminated> Main [Java Application] C:\Users\kopo\p2\p

3
123
23.1
1.7320508075688772
0.9917838735652125

TimeZone Class

TimeZone Class

- ▶ Perform mathematical tasks on numbers

```
12 public static void main(String[] args) {  
13     // TODO Auto-generated method stub  
14     TimeZone time;  
15     Date date = new Date();  
16     SimpleDateFormat dateFormat = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");  
17  
18     time = TimeZone.getTimeZone("Asia/Seoul");  
19     dateFormat.setTimeZone(time);  
20     System.out.println(time.getDisplayName() + " : " + dateFormat.format(date));  
21  
22     time = TimeZone.getTimeZone("America/New_York");  
23     dateFormat.setTimeZone(time);  
24     System.out.println(time.getDisplayName() + " : " + dateFormat.format(date));  
25 }
```

Problems @ Javadoc Declaration Console X

<terminated> Main [Java Application] C:\Users\wkopo\p2\p2

대한민국 표준시 : 2021-02-11 16:45:37
미 동부 표준시 : 2021-02-11 02:45:37

P9 : add "z Z" at the end of dateFormat and print the time again

String Class

String Class

```
12 public static void main(String[] args) {  
13     // TODO Auto-generated method stub  
14     String text = "Hello,hi,";  
15     System.out.println(text);  
16     System.out.println(text.length());  
17     System.out.println(text.charAt(2));  
18  
19     char[] c = text.toCharArray();  
20     System.out.println(c[0]);  
21  
22     System.out.println(text.replace("H", "h"));  
23  
24     String[] t = text.split(",");  
25     System.out.println(t[0]);  
26     System.out.println(t[1]);  
27     System.out.println(t[2]);  
28 }
```

P10 : Explain the reason why this exception occurred

Problems @ Javadoc Declaration Console Coverage
<terminated> Main [Java Application] C:\Users\kopo\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_15.0.1
Hello,hi,
9
1
H
hello,hi,
Hello
hi
Exception in thread "main" [java.lang.ArrayIndexOutOfBoundsException](#): Index 2 out of bounds for length 2
at test.Main.main(Main.java:27)

P11

Currency Converter

- ▶ Print menu for currency conversion
- ▶ User selects a number from the menu
- ▶ User input a number
- ▶ Print the converted currency
- ▶ The menu should have at least three more options

#Current Converter

1. South Korean won -> United States Dollar

2. United States Dollar -> South Korean won

3.

1

1500

1.36

P12

Temperature Converter

- ▶ Print menu for temperature conversion
- ▶ User selects a number from the menu
- ▶ User input a number
- ▶ Print the converted temperature

#Current Converter

1. Celsius -> Fahrenheit

2. Fahrenheit -> Celsius

1

20

68

Extracting Word from String

- ▶ User inputs one sentence and one word
- ▶ Calculate how many input words can be made from the sentence
- ▶ No need to change the font color as below, it just helps you understand

I go to school by subway but it takes a very long time to go to school.

cool

2

P14

Article before Consonant and Vowel

- ▶ User inputs a sentence including at least an article
- ▶ Check if the articles before consonants and vowels are correct
- ▶ Modify the article if needed

This is a apple.
This is an apple.

This is a school.
This is a school.

Numeric Class

Numeric Class

```
13 public static void main(String[] args) {  
14     // TODO Auto-generated method stub  
15     String text = "-31.5123121231231212312315342137";  
16     System.out.println(text);  
17     System.out.println(Float.parseFloat(text));  
18     System.out.println(Double.parseDouble(text));  
19     Double d = Double.parseDouble(text);  
20     System.out.println(d);  
21     System.out.println(d.intValue());  
22     BigDecimal bd = new BigDecimal(text);  
23     System.out.println(bd);  
24     System.out.println(Math.round(d));  
25     System.out.println(Math.floor(d));  
26     System.out.println(Math.ceil(d));  
27     float f = 3.26f;  
28     System.out.println(f);  
29     System.out.println((int)f);  
30 }
```

Problems @ Javadoc Declaration Console

<terminated> Main [Java Application] C:\Users\kopo\p2\p

```
-31.5123121231231212312315342137  
-31.512312  
-31.51231212312312  
-31.51231212312312  
-31  
-31.5123121231231212312315342137  
-32  
-32.0  
-31.0  
3.26  
3
```

P15

Calculator

- ▶ User inputs an equation
- ▶ Print the calculated result
- ▶ Hint, `String.substring()`, `String.contains()`

5*7.5

37.5

32/2

16