

Due to the print book page limit, we cannot include all good CheckPoint questions in the physical book. The CheckPoint on this Website may contain extra questions not printed in the book. The questions in some sections may have been reordered as a result. Nevertheless, it is easy to find the CheckPoint questions in the book on this Website. Please send suggestions and errata to Dr. Liang at y.daniel.liang@gmail.com. Indicate the book, edition, and question number in your email.

Thanks!

Chapter 4 Check Point Questions

Section 4.2

▼ 4.2.1

Evaluate the following method calls:

- (a) `Math.sqrt(4)`
- (b) `Math.sin(2 * Math.PI)`
- (c) `Math.cos(2 * Math.PI)`
- (d) `Math.pow(2, 2)`
- (e) `Math.log(Math.E)`
- (f) `Math.exp(1)`
- (g) `Math.max(2, Math.min(3, 4))`
- (h) `Math rint(-2.5)`
- (i) `Math.ceil(-2.5)`
- (j) `Math.floor(-2.5)`
- (k) `Math.round(-2.5f)`
- (l) `Math.round(-2.5)`
- (m) `Math rint(2.5)`
- (n) `Math.ceil(2.5)`
- (o) `Math.floor(2.5)`
- (p) `Math.round(2.5f)`
- (q) `Math.round(2.5)`
- (r) `Math.round(Math.abs(-2.5))`

- (a) `Math.sqrt(4) = 2.0`
- (b) `Math.sin(2*Math.PI) = 0`
- (c) `Math.cos(2*Math.PI) = 1`
- (d) `Math.pow(2, 2) = 4.0`
- (e) `Math.log(Math.E) = 1.0`
- (f) `Math.exp(1) = 2.718`
- (g) `Math.max(2, Math.min(3, 4)) = 3`
- (h) `Math.rint(-2.5) = -2.0`
- (i) `Math.ceil(-2.5) = -2.0`
- (j) `Math.floor(-2.5) = -3.0`
- (k) `Math.round(-2.5f) = -2`
- (l) `Math.round(-2.5) = -2`
- (m) `Math.rint(2.5) = 2.0`
- (n) `Math.ceil(2.5) = 3.0`
- (o) `Math.floor(2.5) = 2.0`
- (p) `Math.round(2.5f) = 3`
- (q) `Math.round(2.5) = 3`
- (r) `Math.round(Math.abs(-2.5)) = 3`

Hide Answer

▼ 4.2.2

True or false? The argument for trigonometric methods is an angle in radians.

True

Hide Answer

▼ 4.2.3

Write a statement that converts 47 degrees to radians and assigns the result to a variable.

```
double radians = Math.toRadians(47);
```

Hide Answer

▼ 4.2.4

Write a statement that converts $\pi / 7$ to an angle in degrees and assigns the result to a variable.

```
double degree = Math.toDegrees(Math.PI / 7);
```

Hide Answer

▼ 4.2.5

Write an expression that obtains a random integer between 34 and 55. Write an expression that obtains a random integer between 0 and 999. Write an expression that obtains a random number between 5.5 and 55.5.

- (a) `34 + (int)(Math.random() * (55 - 34))`
- (b) `(int)(Math.random() * 1000)`
- (c) `5.5 + (Math.random() * (55.5 - 5.5))`

Hide Answer

▼ 4.2.6

Why does the Math class not need to be imported?

The Math class is in the java.lang package. Any class in the java.lang package is automatically imported. So there is no need to import it explicitly.

Hide Answer

▼ 4.2.7

What is `Math.log(Math.exp(5.5))`? What is `Math.exp(Math.log(5.5))`? What is `Math.asin(Math.sin(Math.PI / 6))`? What is `Math.sin(Math.asin(Math.PI / 6))`?

```
5.5
5.5
0.5235987755982988
0.5235987755982988
```

Hide Answer

Section 4.3

▼ 4.3.1

Use print statements to find out the ASCII code for 'l', 'A', 'B', 'a', and 'b'. Use print statements to find out the character for the decimal codes 40, 59, 79, 85, and 90. Use

print statements to find out the character for the hexadecimal code 40, 5A, 71, 72, and 7A.

```
System.out.println((int)'1');
System.out.println((int)'A');
System.out.println((int)'B');
System.out.println((int)'a');
System.out.println((int)'b');

System.out.println((char)40);
System.out.println((char)59);
System.out.println((char)79);
System.out.println((char)85);
System.out.println((char)90);

System.out.println((char)0X40);
System.out.println((char)0X5A);
System.out.println((char)0X71);
System.out.println((char)0X72);
System.out.println((char)0X7A);
```

Hide Answer

▼ 4.3.2

Which of the following are correct literals for characters?

'1', '\u345dE', '\u3fFa', '\b', '\t'

'\u345dE' is wrong. It must have exactly four hex numbers in the Unicode.

Hide Answer

▼ 4.3.3

How do you display the characters \ and "?

\" and \"

Hide Answer

▼ 4.3.4

Evaluate the following:

```
int i = '1';
int j = '1' + '2' * ('4' - '3') + 'b' / 'a';
int k = 'a';
char c = 90;
```

i is 49, since the ASCII code of '1' is 49.

j is 100.

k is 97 since the ASCII code of 'a' is 97.

c is character 'Z' since (int) 'Z' is 90.

Hide Answer

▼ 4.3.5

Can the following conversions involving casting be allowed? If so, find the converted result.

```

char c = 'A';
int i = (int)c;

float f = 1000.34f;
int i = (int)f;

double d = 1000.34;
int i = (int)d;

int i = 97;
char c = (char)i;

char c = 'A';
i = (int)c; // i becomes 65

float f = 1000.34f;
int i = (int)f; // i becomes 1000

double d = 1000.34;
int i = (int)d; // i becomes 1000

int i = 97;
char c = (char)i; // c becomes 'a'

```

Hide Answer

▼4.3.6

Show the output of the following program:

```

public class Test {
    public static void main(String[] args) {
        char x = 'a';
        char y = 'c';

        System.out.println(++x);
        System.out.println(y++);
        System.out.println(x - y);
    }
}

```

```

b
c
-2

```

Hide Answer

▼4.3.7

Write the code that generates a random lowercase letter.

```

(int)(Math.random() * 26 + 'a')

```

Hide Answer

▼4.3.8

Show the output of the following statements:

```
System.out.println('a' < 'b');
System.out.println('a' <= 'A');
System.out.println('a' > 'b');
System.out.println('a' >= 'A');
System.out.println('a' == 'a');
System.out.println('a' != 'b');
```

```
true
false
false
true
true
true
```

Hide Answer

Section 4.4

▼ 4.4.1

Suppose that s1, s2, and s3 are three strings, given as follows:

```
String s1 = "Welcome to Java";
String s2 = "Programming is fun";
String s3 = "Welcome to Java";
```

What are the results of the following expressions?

- a. s1 == s2
- b. s2 == s3
- c. s1.equals(s2)
- d. s1.equals(s3)
- e. s1.compareTo(s2)
- f. s2.compareTo(s3)
- g. s2.compareTo(s2)
- h. s1.charAt(0)
- i. s1.indexOf('j')
- j. s1.indexOf("to")
- k. s1.lastIndexOf('a')
- l. s1.lastIndexOf("o", 15)
- m. s1.length()
- n. s1.substring(5)
- o. s1.substring(5, 11)
- p. s1.startsWith("Wel")
- q. s1.endsWith("Java")
- r. s1.toLowerCase()
- s. s1.toUpperCase()
- t. s1.concat(s2)
- u. s1.contains(s2)
- v. "\t Wel \t".trim()

- a. false
- b. false
- c. false
- d. true
- e. a positive number
- f. a negative number
- g. 0
- h. W

- i. -1
- j. 8
- k. 14
- l. 9
- m. 15
- n. me to Java
- o. me to
- p. true
- q. true
- r. welcome to java
- s. WELCOME TO JAVA
- t. Welcome to JavaProgramming is fun
- u. false
- v. Wel

Hide Answer

▼ 4.4.2

Suppose that s1 and s2 are two strings. Which of the following statements or expressions are incorrect?

```
String s = "Welcome to Java";
String s3 = s1 + s2;
String s3 = s1 - s2;
s1 == s2;
s1 >= s2;
s1.compareTo(s2);
int i = s1.length();
char c = s1(0);
char c = s1.charAt(s1.length());
```

String s = "Welcome to Java";
Answer: Correct

String s3 = s1 + s2;
Answer: Correct

String s3 = s1 - s2;
Answer: Incorrect

s1 == s2
Answer: Correct

s1 >= s2
Answer: Incorrect

s1.compareTo(s2);
Answer: Correct

int i = s1.length();
Answer: Correct

char c = s1(0);
Answer: Incorrect

char c = s1.charAt(s1.length());

Answer: Incorrect : it's out of bounds, even if the preceding problem is fixed.

Hide Answer

▼ 4.4.3

Show the output of the following statements (write a program to verify your results):

```
System.out.println("1" + 1);
System.out.println('1' + 1);
System.out.println("1" + 1 + 1);
System.out.println("1" + (1 + 1));
System.out.println('1' + 1 + 1);

System.out.println("1" + 1); => 11
System.out.println('1' + 1); => 50
// (since the Unicode for 1 is 49
System.out.println("1" + 1 + 1); => 111
System.out.println("1" + (1 + 1)); => 12
System.out.println('1' + 1 + 1); => 51
```

Hide Answer

▼ 4.4.4

Evaluate the following expressions (write a program to verify your results):

```
1 + "Welcome " + 1 + 1
1 + "Welcome " + (1 + 1)
1 + "Welcome " + ('\u0001' + 1)
1 + "Welcome " + 'a' + 1

1 + "Welcome " + 1 + 1 is 1Welcome 11.
1 + "Welcome " + (1 + 1) is 1Welcome 2.
1 + "Welcome " + ('\u0001' + 1) is 1Welcome 2
1 + "Welcome " + 'a' + 1 is 1Welcome a1
```

Hide Answer

▼ 4.4.5

Let s1 be " Welcome " and s2 be " welcome ". Write the code for the following statements:

- Check whether s1 is equal to s2 and assign the result to a Boolean variable isEqual.
- Check whether s1 is equal to s2, ignoring case, and assign the result to a Boolean variable isEqual.
- Compare s1 with s2 and assign the result to an int variable x.
- Compare s1 with s2, ignoring case, and assign the result to an int variable x.
- Check whether s1 has the prefix AAA and assign the result to a Boolean variable b.
- Check whether s1 has the suffix AAA and assign the result to a Boolean variable b.
- Assign the length of s1 to an int variable x.
- Assign the first character of s1 to a char variable x.
- Create a new string s3 that combines s1 with s2.
- Create a substring of s1 starting from index 1.
- Create a substring of s1 from index 1 to index 4.
- Create a new string s3 that converts s1 to lowercase.
- Create a new string s3 that converts s1 to uppercase.
- Create a new string s3 that trims whitespaces on both ends of s1.

- o. Assign the index of the first occurrence of the character e in s1 to an int variable x.
p. Assign the index of the last occurrence of the string abc in s1 to an int variable x.

```
(a) boolean isEqual = s1.equals(s2);
(b) boolean isEqual = s1.equalsIgnoreCase(s2);
(c) int x = s1.compareTo(s2);
(d) int x = s1.compareToIgnoreCase(s2);
(e) boolean b = s1.startsWith("AAA");
(f) boolean b = s1.endsWith("AAA");
(g) int x = s1.length();
(h) char x = s1.charAt(0);
(i) String s3 = s1 + s2;
(j) String s3 = s1.substring(1);
(k) String s3 = s1.substring(1, 5);
(l) String s3 = s1.toLowerCase();
(m) String s3 = s1.toUpperCase();
(n) String s3 = s1.trim();
(o) int x = s1.indexOf('e');
(p) int x = s1.lastIndexOf("abc");
```

Hide Answer

▼ 4.4.6

Write one statement to return the number of digits in an integer i.

```
(i + "").length()
```

Hide Answer

▼ 4.4.7

Write one statement to return the number of digits in a double value d.

```
(d + "").length()
```

Hide Answer

▼ 4.4.8

What is wrong in the following code?

```
import java.util.Scanner;

public class Test {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print("Enter an integer: ");
        int value = input.nextInt();
        System.out.println("The value is " + value);

        System.out.print("Enter a line: ");
        String line = input.nextLine();
        System.out.println("The line is " + line);
    }
}
```

input.nextLine() is used after input.nextInt(). Don't use a line-based input after a token-based input.

Hide Answer

Section 4.5

▼ 4.5.1

If you run Listing 4.3 GuessBirthday.java with input 1 for Set1, Set3, and Set4 and 0 for Set2 and Set5, what will be the birthday?

13

Hide Answer

▼ 4.5.2

If you enter a lowercase letter such as b, the program in Listing 4.4 displays B is 11. Revise the code so to display b is 11.

Change ch in lines 20, 24, 27 to `hexString.charAt(0)`.

Hide Answer

▼ 4.5.3

What would be wrong if lines 6-7 in Listing 4.5 is replaced by the following code?
`String lottery = "" + (int)(Math.random() * 100);`

lottery may have only one digit.

Hide Answer

Section 4.6

▼ 4.6.1

What are the format specifiers for outputting a Boolean value, a character, a decimal integer, a floating-point number, and a string?

The specifiers for outputting a boolean value, a character, a decimal integer, a floating-point number, and a string are `%b`, `%c`, `%d`, `%f`, and `%s`.

Hide Answer

▼ 4.6.2

What is wrong in the following statements?

- (a) `System.out.printf("%5d %d\n", 1, 2, 3);`
- (b) `System.out.printf("%5d %f\n", 1);`
- (c) `System.out.printf("%5d %f\n", 1, 2);`
- (d) `System.out.printf("%.2f\n%0.3f\n", 1.23456, 2.34);`
- (e) `System.out.printf("%08s\n", "Java");`

- (a) the last item 3 does not have any specifier.
- (b) There is not enough items.
- (c) The data for `%f` must a floating-point value.
- (d) `%0.3f` is wrong. Width cannot be zero.
- (e) `%08s` is wrong. 0 should be removed.

Hide Answer

▼ 4.6.3

Show the output of the following statements.

- (a) `System.out.printf("amount is %f %e\n", 32.32, 32.32);`
- (b) `System.out.printf("amount is %5.2f%% %5.4e\n", 32.327, 32.32);`
- (c) `System.out.printf("%6b\n", (1 > 2));`
- (d) `System.out.printf("%6s\n", "Java");`
- (e) `System.out.printf("%-6b%s\n", (1 > 2), "Java");`
- (f) `System.out.printf("%6b%-8s\n", (1 > 2), "Java");`
- (g) `System.out.printf("%,5d %,6.1f\n", 312342, 315562.932);`
- (h) `System.out.printf("%05d %06.1f\n", 32, 32.32);`

(a) amount is 32.320000 3.233000e+01

(b) amount is 32.33% 3.2330e+01

(c) *false // * denote a space

(d) **Java // * denote a space

(e) false*Java

(f) *falseJava****

(g) 312,342 315,562.9

(h) 00032 0032.3

Hide Answer