**Computer Language 2023**

**Assignment #3**

**Due: 26/Mar 23:59:59**

**Student ID:\_21102054\_ Name: \_Lim Saeyeon (임새연)\_**

**For each question, 1) write your solution codes, 2) present a screenshot of your result, and 3) describe a short explanation about your solution. Without these components, you will be given some penalties.**

**1. Convert the following if-else statement to switch-case statement.**

**public static void** main(String[] args) {  
  
 **int** i = 0;  
 **if** (i == 1)  
 System.***out***.println(**"!"**);  
 **else if** (i == 2)  
 System.***out***.println(**"@"**);  
 **else if** (i == 3)  
 System.***out***.println(**"#"**);  
 **else** System.***out***.println(**"\*"**);  
  
}

**1) Your code:**

public class question1 {  
 public static void main(String[] args) {  
  
 int i = 0;  
  
 switch (i) {  
 case 1:  
 System.*out*.println("!");  
 case 2:  
 System.*out*.println("@");  
 case 3:  
 System.*out*.println("#");  
 default:  
 System.*out*.println("\*");  
 }  
 }  
}

**2) Your result (screenshot):**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**3) Your explanation:**

I used a switch statement to print out different characters based on the value of the variable "i". Since "i" is initialized to 0, it will not match any of the cases and will execute the default case, printing out an asterisk. If "i" were set to 1, the program would print out an exclamation mark, and so on for values 2 and 3.

Next page >>

**2. Write a program that takes 2digit positive integer (10~99) from the user, and then checks if ten’s digit number and one’s digit number are same or not.**

텍스트이(가) 표시된 사진

자동 생성된 설명텍스트이(가) 표시된 사진

자동 생성된 설명

**1) Your code:**

import java.util.Scanner;  
  
public class question2 {  
 public static void main(String[] args) {  
  
 Scanner sc = new Scanner(System.*in*);  
 System.*out*.print("Input 2-digit number (10~99): ");  
 int num = sc.nextInt();  
  
 if (num < 10 || num > 99) {  
 System.*out*.println("Wrong input!");  
 } else if (num / 10 == num % 10) {  
 System.*out*.println("Yes! two numbers are same!");  
 } else {  
 System.*out*.println("No! two numbers are NOT same!");  
 }  
 sc.close();  
 }  
}

**2) Your result (screenshot)**

**텍스트이(가) 표시된 사진

자동 생성된 설명 텍스트이(가) 표시된 사진

자동 생성된 설명**

**3) Your explanation:**

The user will be asked to input a two-digit number between 10 and 99. If the input is outside of this range, the program prints "Wrong input!". If the input is within the range, the program checks if the two digits in the input are the same. If they are the same, the program prints "Yes! two numbers are same!". If they are not the same, the program prints "No! two numbers are NOT same!". I used the Scanner class to read user input and the if-else statements to control the flow of the program based on the input. Finally, the Scanner is closed to free up system resources.

**3. Write a simple calculator program. The available arithmetic operators are +, -, \*, and / operators. Operands and operator must be separated by whitespaces. The program must print “Cannot divide by 0” string when a user tries to “divide by 0” operation. Use a switch-case statement.**

텍스트이(가) 표시된 사진

자동 생성된 설명텍스트이(가) 표시된 사진

자동 생성된 설명텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트, 장치, 게이지이(가) 표시된 사진

자동 생성된 설명텍스트, 장치이(가) 표시된 사진

자동 생성된 설명

**1) Your code:**

import java.util.Scanner;  
  
public class question3 {  
 public static void main(String[] args) {  
 Scanner sc = new Scanner(System.*in*);  
  
 System.*out*.print("Operation >>");  
 int num1 = sc.nextInt();  
 String operator = sc.next();  
 int num2 = sc.nextInt();  
  
 switch (operator) {  
 case "+" :  
 System.*out*.println(num1 + "+" + num2 + "= " + (num1 + num2));  
 break;  
 case "-" :  
 System.*out*.println(num1 + "-" + num2 + "= " + (num1 - num2));  
 break;  
 case "\*" :  
 System.*out*.println(num1 + "\*" + num2 + "= " + (num1 \* num2));  
 break;  
 case "/" : {  
 if (num2 == 0) {  
 System.*out*.println("Cannot divide by 0");  
 } else {  
 System.*out*.println(num1 + "/" + num2 + "= " + (num1 / num2));  
 }  
 }  
 break;  
 default :  
 System.*out*.println("Invalid operator");  
 }  
  
 sc.close();  
 }  
}

**2) Your result (screenshot)**

**텍스트이(가) 표시된 사진

자동 생성된 설명 텍스트이(가) 표시된 사진

자동 생성된 설명**

**텍스트이(가) 표시된 사진

자동 생성된 설명 텍스트이(가) 표시된 사진

자동 생성된 설명**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**3) Your explanation:**

I made a simple calculator that takes in two numbers and an operator (+, -, \*, /) from the user, and this performs the arithmetic operation on the two numbers. I used the Scanner class to read user input and the switch statement to determine which operation to perform based on the input operator.

The user should enter the first number, followed by the operator, and then the second number. Then, this program uses a switch statement to check which operator was entered and perform the operation. If the user entered an invalid operator, the program prints "Invalid operator".

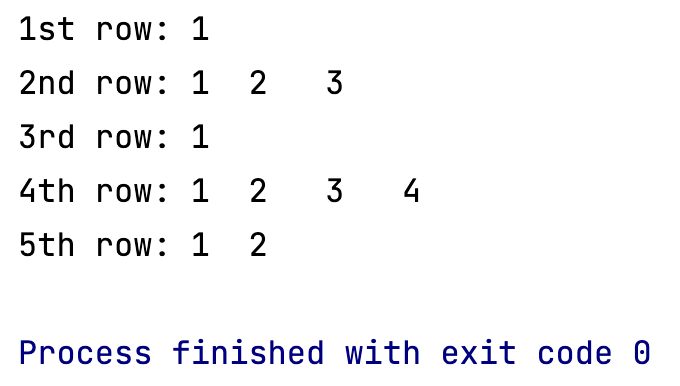
If the operator entered is "/", the program first checks if the second number is 0. If it is, the program prints "Cannot divide by 0". If it is not, the program performs the division operation and prints the result. Finally, the Scanner is closed to free up system resources.

Next page >>

**4. Write a program that prints out the following 2d array:**

**int n [][] = { {1}, {1,2,3}, {1}, {1,2,3,4}, {1,2}};**

**\* use switch-case statement for printing the row number (i.e., 1st, 2nd, … 5th)!**



**1) Your code:**

public class question4 {  
 public static void main(String[] args) {  
  
 int n[][] = {{1}, {1, 2, 3}, {1}, {1, 2, 3, 4}, {1, 2}};  
  
 for (int i = 1; i <= 5; i++) {  
 switch(i) {  
 case 1:  
 System.*out*.print(i + "st row: ");  
 for (int j = 0; j < n[0].length; j++) {  
 System.*out*.print(n[0][j] + "\t");  
 }  
 break;  
 case 2:  
 System.*out*.print(i + "nd row: ");  
 for (int j = 0; j < n[1].length; j++) {  
 System.*out*.print(n[1][j] + "\t");  
 }  
 break;  
 case 3:  
 System.*out*.print(i + "rd row: ");  
 for (int j = 0; j < n[2].length; j++) {  
 System.*out*.print(n[2][j] + "\t");  
 }  
 break;  
 case 4:  
 System.*out*.print(i + "th row: ");  
 for (int j = 0; j < n[3].length; j++) {  
 System.*out*.print(n[3][j] + "\t");  
 }  
 break;  
 case 5:  
 System.*out*.print(i + "th row: ");  
 for (int j = 0; j < n[4].length; j++) {  
 System.*out*.print(n[4][j] + "\t");  
 }  
 break;  
 default:  
 System.*out*.println("Invalid row number!");  
 break;  
 }  
 System.*out*.println();  
 }  
 }  
}

**2) Your result (screenshot)**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**3) Your explanation:**

This program defines a 2D array n with five rows, where each row contains a different number of elements. I used a for loop to iterate through the rows and a switch statement to print out each row's contents in a specific format.

The program first initializes the 2D array n with the specified values. The for loop iterates through the row numbers from 1 to 5, and the switch statement is used to print out each row's contents in a specific format.

For each row number, the switch statement checks which case it matches and prints the corresponding message. For example, for the first row (i=1), the program prints "1st row: " and then uses a nested for loop to iterate through the elements in the first row of the array and print them out separated by tabs. The other cases for the remaining rows are similar. If the row number entered is not between 1 and 5, the program prints "Invalid row number!" and continues to the next iteration of the loop.

Finally, System.out.println() is used to move to the next line after printing the contents of each row.

**5. Write a simple program that takes a decimal system number and print the octal system number. Assume that a user will type 0~1000 as input number. You cannot use Integer class’s method to solve this question.**

**텍스트이(가) 표시된 사진

자동 생성된 설명 텍스트이(가) 표시된 사진

자동 생성된 설명텍스트이(가) 표시된 사진

자동 생성된 설명**

**1) Your code:**

import java.util.Scanner;  
  
public class question5 {  
 public static void main(String[] args) {  
  
 Scanner sc = new Scanner(System.*in*);  
 System.*out*.print("Input your number (decimal): ");  
 int num = sc.nextInt();  
 System.*out*.println("Your number in Octal system: " + String.*format*("%o", num));  
 sc.close();  
 }  
}

**2) Your result (screenshot)**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**3) Your explanation:**

This program takes an integer input in decimal format from the user using the Scanner class and then converts it into an octal number. The program then prints the converted octal number to the console.

The program first prompts the user to enter an integer number in decimal format. The Scanner class is used to take the user's input.

Next, the program converts the decimal number into an octal number using the String.format() method. The "%o" format specifier is used to indicate that the input number should be converted into an octal number.

Finally, I used System.out.println() to print the converted octal number to the console along with an informative message. Then, scanner is closed using the sc.close() method to prevent resource leaks.

Next page >>

**6. Write a program to print out the following triangle pattern. The program should take the height of a triangle from the user (user’s input will be 3~10). Hardcoded program is not accepted.**

**테이블이(가) 표시된 사진

자동 생성된 설명텍스트이(가) 표시된 사진

자동 생성된 설명**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**1) Your code:**

import java.util.Scanner;  
  
public class question6 {  
 public static void main(String[] args) {  
 Scanner sc = new Scanner(System.*in*);  
 System.*out*.print("Enter the number of rows: ");  
 int num = sc.nextInt();  
 if (num < 3 || num > 10) {  
 System.*out*.println("Wrong input");  
 return;  
 }  
 for (int i = num; 0 < i; i--) {  
 for (int j = i; 0 < j; j--) {  
 System.*out*.print("\*");  
 }  
 System.*out*.println();  
 }  
 sc.close();  
 }  
}

**텍스트이(가) 표시된 사진

자동 생성된 설명텍스트이(가) 표시된 사진

자동 생성된 설명2) 텍스트이(가) 표시된 사진

자동 생성된 설명Your result (screenshot)**

**3) Your explanation:**

The user will be asked to input the number of rows they want to generate for a pyramid made of asterisks. If the input is less than 3 or greater than 10, the program outputs "Wrong input" and exits the program. If the input is valid, the program generates the pyramid by printing asterisks in descending order for each row until the desired number of rows is reached.

The program first prompts the user to enter the number of rows they want to generate for the pyramid. The Scanner class is used to take the user's input. Then, the program checks if the input is less than 3 or greater than 10. If so, it outputs "Wrong input" and exits the program using the return statement. If the input is valid, the program generates the pyramid by using two nested for loops. The outer loop iterates through each row from the total number of rows down to 1. The inner loop prints asterisks for each column in the row, from the current row number down to 1.

Finally, the program uses System.out.println() to print a new line character at the end of each row to start a new line. At the end of the program, the Scanner is closed using the sc.close() method to prevent resource leaks.

**7. Write a program to print out the following triangle pattern. The program should take the height of a triangle from the user (user’s input will be 3~10). Hardcoded program is not accepted.**

**테이블이(가) 표시된 사진

자동 생성된 설명텍스트이(가) 표시된 사진

자동 생성된 설명**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**1) Your code:**

import java.util.Scanner;  
  
public class question7 {  
 public static void main(String[] args) {  
 Scanner sc = new Scanner(System.*in*);  
 System.*out*.print("Enter the number of rows: ");  
 int num = sc.nextInt();  
 if (num < 3 || num > 10) {  
 System.*out*.println("Wrong input");  
 return;  
 }  
 for (int i = num; 0 < i; i--) {  
 if (i == num || i <= 2) {  
 for (int j = i; 0 < j; j--) {  
 System.*out*.print("\*");  
 }  
 } else {  
 System.*out*.print("\*");  
 for (int j = i - 2; 0 < j; j--) {  
 System.*out*.print(" ");  
 }  
 System.*out*.print("\*");  
 }  
 System.*out*.println();  
 }  
 sc.close();  
 }  
}

**텍스트이(가) 표시된 사진

자동 생성된 설명2) 텍스트이(가) 표시된 사진

자동 생성된 설명Your result (screenshot)**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**3) Your explanation:**

I can print a pattern of ‘\*’ based on the user's input for the number of rows by using this program. The pattern consists of rows that are shaped like a triangle, where the top and bottom rows are complete lines of ‘\*’, and the middle rows are partially filled with ‘\*’.

The program first prompts the user to enter the number of rows they want to print. If the user enters a number less than 3 or greater than 10, the program prints "Wrong input" and exits. Otherwise, the program enters a loop that iterates from the user-input number down to 1.

Within the loop, the program checks the current row number “i”. If i is equal to the user-input number or less than or equal to 2, the program prints a complete row of ‘\*’. Otherwise, the program prints a partially filled row with ‘\*’ at the beginning and end, and spaces in between. After each row is printed, the program goes to the next line. Finally, the program closes the scanner object.

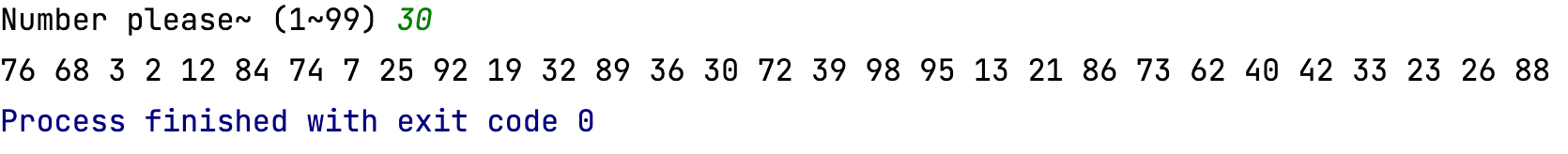
Next page >>

**8. Write a program that 1) takes a positive integer number *N* (less than 100) from the user, 2) creates an integer array with the size of *N,* 3) fills in this array using a set of random numbers ranging from 1~100, 4) prints out the numbers in this array. However, this array CANNOT have the duplicated numbers.**

**\* Hint: find the usage of Math.random() method**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

****

**1) Your code:**

import java.util.Scanner;  
import java.util.HashSet;  
  
public class question8 {  
 public static void main(String[] args) {  
 Scanner sc = new Scanner(System.*in*);  
  
 System.*out*.print("Number please~ (1~99) ");  
 int num = sc.nextInt();  
  
 if (num < 1 || num > 99) {  
 System.*out*.println("Wrong input!");  
 return;  
 }  
 int[] arr = new int[num];  
  
 HashSet<Integer> set = new HashSet<Integer>();  
  
 for (int i = 0; i < num; i++) {  
 int number = (int) (Math.*random*() \* 100) + 1;  
 while (set.contains(number)) {  
 number = (int) (Math.*random*() \* 100) + 1;  
 }  
 arr[i] = number;  
 set.add(number);  
 }  
  
 for (int i = 0; i < num; i++) {  
 System.*out*.print(arr[i] + " ");  
 }  
 System.*out*.println();  
 sc.close();  
 }  
}

**2) Your result (screenshot)**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**3) Your explanation:**

This program generates an array of random unique integers between 1 and 99, based on the user's input of the desired size of the array. It starts by prompting the user to input a number between 1 and 99, and if the input is out of range, it outputs an error message and exits the program. Then it creates an integer array of size num and a HashSet to store unique integers. It uses a while loop to generate random integers and checks if the integer already exists in the HashSet, if so it generates a new random integer until it finds a unique one, which is then added to both the array and the HashSet. Finally, it prints out the generated array.

Next page >>

**9. Write a program that takes a single string from the user. Then, your program needs to convert uppercase letters included in that string into lowercase letters and vice versa. The result should be identical to the following examples. However, you cannot use String/Character class’s toupper()/tolower()-related methods.**

**\*Hint: Find the usage of String’s charAt() method to take the n-th character of the string  
\*Hint: what is the Unicode number of ‘A’ and ‘a’?**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**1) Your code:**

import java.util.Scanner;

public class question9 {  
 public static void main(String[] args) {  
  
 Scanner sc = new Scanner(System.*in*);  
 System.*out*.println("Input your string:");  
 String str = sc.next();  
 char ch;  
  
 for (int i = 0; i < str.length(); i++) {  
 ch = str.charAt(i);  
 if ('A' <= ch && ch <= 'Z') {  
 ch += 32;  
 } else if ('a' <= ch && ch <= 'z') {  
 ch -= 32;  
 }  
 System.*out*.print(ch);  
 }  
 System.*out*.println();  
 sc.close();  
 }  
}

**2) Your result (screenshot)**

**텍스트이(가) 표시된 사진

자동 생성된 설명 텍스트이(가) 표시된 사진

자동 생성된 설명**

**3) Your explanation:**

This program takes a user input string and converts all uppercase characters to lowercase and all lowercase characters to uppercase. It does this by iterating over each character in the string and checking if it is an uppercase or lowercase letter. If it is uppercase, it adds 32 to its Ascii value to convert it to lowercase. If it is lowercase, it subtracts 32 from its Ascii value to convert it to uppercase.