JOBSHEET 10
Array 2

1. Objective

- Students are able to understand how to create 2-dimensional arrays in the Java programming language
- Students are able to access 2-dimensional array

2. Laboratory

2.1 Experiment 1: Declare, Initialize, and Display 2-Dimensional Array

Experiment Time: 60 minutes

In this experiment, program code was created to declare, initialize, and display elements in a 2-dimensional array. The data stored is the name of the mini cinema audience who will be seated in the room with 4 rows and 2 columns of seats.

1. Open a text editor. Create a new file, name it **Cinema<StudentID>.java**
2. Write the basic structure of the Java programming language which contains the **main()** function
3. Create an array of String type named **audience** with a row capacity of 4 elements and a column of 2 elements

```
String[][] audience = new String[4][2];
```

4. Fill in each element of the value array as follows:

```
audience[0][0] = "Amin";
audience[0][1] = "Bena";
audience[1][0] = "Candra";
audience[1][1] = "Dela";
audience[2][0] = "Eka";
audience[2][1] = "Farhan";
audience[3][0] = "Gisel";
```

5. Display all contents of the elements to the screen

```
System.out.printf("%s \t %s\n", audience[0][0], audience[0][1]);
System.out.printf("%s \t %s\n", audience[1][0], audience[1][1]);
System.out.printf("%s \t %s\n", audience[2][0], audience[2][1]);
System.out.printf("%s \t %s\n", audience[3][0], audience[3][1]);
```

6. Compile and run the program. Match the results of the running programs that you have created according to the following display

```
Amin     Bena
Candra   Dela
Eka      Farhan
Gisel    null
```

Questions!

1. Do array elements have to be filled in sequentially starting from the 0th index? Please explain!
2. Why is there a null in the list of audience names?
3. Complete the audience list in step 4 so that it looks like the following program code

```
audience[0][0] = "Amin";
audience[0][1] = "Bena";
audience[1][0] = "Candra";
audience[1][1] = "Dela";
audience[2][0] = "Eka";
audience[2][1] = "Farhan";
audience[3][0] = "Gisel";
audience[3][1] = "Hana";
```

4. Add the following program code:

```
System.out.println(audience.length);
System.out.println(audience[0].length);
System.out.println(audience[1].length);
System.out.println(audience[2].length);
System.out.println(audience[3].length);
```

   Explain the function of audience.length and audience[0].length!
   Do audience[0].length, audience[1].length, audience[2].length, and audience[3].length have the same value? Why?
5. Modify the program code in step 4 to display the length of each row in the array using a for loop. Compile, run, then commit.

```
System.out.println(audience.length);
for (int i = 0; i < audience.length; i++) {
    System.out.println("Length of row " + (i + 1) + ": " + audience[i].length);
}
```

6. Modify the program code in step 5 to display the length of each row in the array using a foreach loop. Compile, run, then commit.

```
for (String[] rowAudience : audience) {
    System.out.println("Length of row: " + rowAudience.length);
}
```

7. In your opinion, what are the advantages and disadvantages of foreach loop compared to for loop?
8. What is the max row index for the audience array?
9. What is the max column index for the audience array?
10. Add program code to display the audience's name on the 3rd line using a for loop. Compile, run, then commit.

```
System.out.println(x:"Audiences in the row 3: ");
for (int i = 0; i < audience[2].length; i++) {
    System.out.println(audience[2][i]);
}
```

11. Modify the code in question number 10 to repeat using a foreach loop. Compile, run, then commit.

```
System.out.println(x:"Audiences in the row 3: ");
for (String i : audience[2]) {
    System.out.println(i);
}
```

12. Modify the program code in question number 11 again to display the audience's name for each line. Compile and run the program then observe the results, then commit.

```
for (int i = 0; i < audience.length; i++) {
    System.out.println("Audience in the row: " + (i + 1) + String.join( ", ", audience[i]));
}
```

13. What is the function of String.join()?
14. Commit and push to GitHub

2.2 Experiment 2: Utilizing Scanners and Loops for Input and Output on 2-Dimensional Arrays

Experiment Time: 80 minutes

Experiment 2 is a follow-up experiment to Experiment 1. In this experiment, a program code was created to store data on the names of spectators in a mini cinema with 4 rows and 2 columns of seats using a scanner.

1. Open a text editor. Create a new file, name it **CinemaWithScanner<StudentID>.java**

2. Write the basic structure of the Java programming language which contains the **main()** function
3. Add Scanner library
4. Declare the Scanner variable
5. Declare row and column variables of type **int** and name and next of type **String**.
6. Create an array of String type named **audience** with a row capacity of 4 elements and a column of 2 elements

```
String[][] audience = new String[4][2];
```

7. Use the scanner to fill in the elements in the **audience** array

```
while (true) {
    System.out.print(s:"Enter a name: ");
    name = sc.nextLine();
    System.out.print(s:"Enter row number: ");
    row = sc.nextInt();
    System.out.print(s:"Enter column number: ");
    column = sc.nextInt();
    sc.nextLine();

    audience[row - 1][column - 1] = name;
    System.out.print(s:"Are there any other audiences to be added? (y/n): ");
    next = sc.nextLine();

    if (next.equalsIgnoreCase(anotherString:"n")) {
        break;
    }
}
```

8. Compile and run the program then try inputting some audience data.

```
Enter a name: Dewi
Enter row number: 1
Enter column number: 2
Are there any other audiences to be added? (y/n): y
Enter a name: Olan
Enter row number: 3
Enter column number: 1
Are there any other audiences to be added? (y/n): y
Enter a name: Gea
Enter row number: 2
Enter column number: 2
Are there any other audiences to be added? (y/n): n
```

9. Commit program code

Questions!

1. Should the array elements from the scanner be filled in sequentially starting from the 0th index? Please explain!

2. Modify the program code to provide the following menu options:
   - Menu 1: Input audience data
   - Menu 2: Show audience list
   - Menu 3: Exit
3. Modify the program code to handle if the seat row/column number is not available
4. In menu 1, modify the program code to give a warning if the selected seat is already occupied by other audiences, then display a command to enter rows and columns again
5. In menu 2, if the seat is empty, replace null with ***
6. Commit and push the program code to GitHub

2.3 Experiment 3: 2-Dimensional Array with Different Row Lengths

Experiment Time: 80 minutes

1. Open a text editor. Create a new file, name it **Numbers<StudentID>.java**
2. Write the basic structure of the Java programming language which contains the **main()** function
3. Declaration and instantiation of a 2-dimensional array named **myNumbers** with elements of type int. The array has 3 rows. The first row consists of 5 columns. The second row consists of 3 columns. The third row consists of 1 column.

```
int[][] myNumbers = new int[3][];
myNumbers[0] = new int[5];
myNumbers[1] = new int[3];
myNumbers[2] = new int[1];
```

4. Commit and push the program code to GitHub

Questions!

1. Add the following program code:

```
for (int i = 0; i < myNumbers.length; i++) {
    System.out.println(Arrays.toString(myNumbers[i]));
}
```

2. What is the function of Arrays.toString()?
3. What is the default value for elements in an array with the data type int?

4. Add the following program code:

```
for (int i = 0; i < myNumbers.length; i++) {
    System.out.println("Length of row " + (i + 1) + ": " + myNumbers[i].length);
}
```

5. The **myNumbers** array has a different length for each row. How to make the length for each row the same? Can the array length be modified?

3. Assignment

Experiment Time: 120 minutes

1. Implement the flowchart that was created in the 10 meeting assignments for the Basic Programming course related to the group project into Java program code. Commit and push your program code to the GitHub repository for the final project Note: assignments may only apply material from meeting 1 to meeting 10