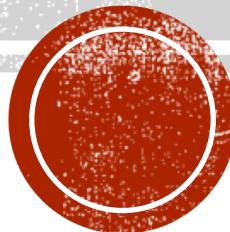




Middlesex  
University  
London

# Lecture 9

## 2D Array



# REVISION TASK

- Write a program containing results from all of your modules which you took so far.
- Populate the array with your results and print it out to the console.
- Return the largest element in an array.



# TODAY:

- Two-dimensional arrays,
- Array creation
- Reading and writing to 2D array
- Programming operations on 2D arrays
- Displaying arrays
- Summing all elements
- Finding min and max elements, and random shuffling.



# INTRODUCTION

- Data in table or a matrix can be represented using a two-dimensional array

Example:

	Q1	Q2	Q3	Q4	Q5
Student1	10	10	20	10	10
Student2	20	20	20	20	20
Student3	20	10	10	20	20
Student4	20	20	10	10	10

The screenshot shows a Java development environment. The code editor displays a class named `ArrayDemo` with a main method that prints a 5x5 matrix of integers. The output window shows the printed matrix and a successful build message.

```
/*
public class ArrayDemo {

    public static void main(String[] args) {

        int [][] table =
        {{10,10,20,10,10},
        {20,20,20,20,20},
        {20,10,10,20,20},
        {20,20,10,10,10}};

        for (int row = 0; row < table.length; row++) {
            System.out.println(" ");
            for (int column = 0; column< table[row].length; column++) {
                System.out.print(" "+table[row][column]);
            }
        }
    }
}
```

Output window content:

```
CST1510 (run-single) #2
run:
10 10 20 10 10
20 20 20 20 20
20 10 10 20 20
20 20 10 10 10
BUILD SUCCESSFUL
```

# WHAT IS 2D ARRAY

- Two dimensional array is used for purpose of storing data in tabular format
- The array is made up of rows and columns.
- The array in Java can have up to 255 dimensions
- 2D array is an array of array, meaning array is stored in another array



# DECLARING 2D ARRAY

```
datatype [ ] [ ] arrayRefVar;
```

Example:

```
int [ ] [ ] table;
```



# CREATING ARRAY

```
arrayRefVar = new datatype [rowIndex] [columnIndex];
```

Example:

```
table = new int [4] [5];
```

rows      columns  
↓            ↗

4 rows      5 columns

	Q1	Q2	Q3	Q4	Q5
Student1	0	0	0	0	0
Student2	0	0	0	0	0
Student3	0	0	0	0	0
Student4	0	0	0	0	0



# DECLARING & CREATING 2D ARRAY

```
datatype [][] arrayRefVar = new datatype[rowI][columnI];
```

Example:

```
int [][] table = new int [4][5];
```



# 2D ARRAY SIZE

- You can find array rows size using

```
arrayRefVar.length
```

- You can find array columns size using

```
arrayRefVar[0].length
```

Examples:

```
table.length;
```

Returns: 4

```
table[0].length;
```

Returns: 5



# INITIALIZE ELEMENTS IN AN ARRAY

```
arrayRefVar[rowIndex] [columnIndex] = value;
```

Example:

rows      columns

```
table [0][0] = 1;  
table [0][1] = 5;  
table [0][2] = 10;  
table [0][3] = 15;  
table [0][4] = 20;  
table [1][0] = 10;  
table [1][1] = 15;  
table [1][2] = 20;  
table [1][3] = 25;  
table [1][4] = 30;  
...
```

rows      columns

		0	1	2	3	4
		Q1	Q2	Q3	Q4	Q5
0	Student1	1	5	10	15	20
1	Student2	10	15	20	25	30
2	Student3	0	0	0	0	0
3	Student4	0	0	0	0	0



# ARRAY INITIALIZER

- A shorthand notation:

```
datatype [][] arrayRefVar = {value, value, value, value};
```

Example:

```
int [][] table =  
    rows {  
        {{ 1, 5, 10, 15, 20},  
         {10, 15, 20, 25, 30},  
         {20, 10, 10, 20, 20},  
         {10, 20, 30, 40, 50}};  
    } columns
```



# RETRIEVE AN ELEMENT FROM AN ARRAY

```
arrayRefVar[rowIndex][columnIndex];
```

Example:

```
System.out.println(table [1][1]);
```

```
int [][] table =  
    {  
        { 0, 1, 2, 3, 4 },  
        { { 1, 5, 10, 15, 20 },  
          { 10, 15, 20, 25, 30 },  
          { 20, 10, 10, 20, 20 },  
          { 10, 20, 30, 40, 50 } }  
    };
```



# RETRIEVE ELEMENTS FROM AN ARRAY

```
for (int row=0; index < arrayRefVar.length; row++) {  
    System.out.println("");  
    for (int column=0; column < arrayRefVar[row].length; column++) {  
        System.out.print(arrayRefVar[row][column]);  
    }  
}
```

Example:

```
for (int row = 0; row < table.length; row++) {  
    System.out.println("");  
    for (int column = 0; column < table[row].length; column++) {  
        System.out.print("\t"+table[row][column]);  
    }  
}
```



# TASK

- Write program returning the largest element from the array.

```
int [][] testResults =  
  
    {{100, 5,10,15,20},  
     {10,15,20,25,30},  
     {20,10,10,20,20},  
     {10,20,30,40,50}};
```



# **KEY THINGS YOU SHOULD UNDERSTAND AFTER WEEK 10**

- Two-dimensional arrays,
- Array creation
- Reading and writing to 2D array
- Programming operations on 2D arrays
- Displaying arrays
- Summing all elements
- Finding min and max elements, and random shuffling.



# MASTERING YOUR SKILLS

- Read chapter 8 from the book and go over the exercises in the end of the chapter.

