



# COMP90041 Programming and Software Development

Tutorial 7 Array

Chuang(Frank) Wang





# Overview

1. Introduction to Array
2. Use Arrays
3. Exercise



THE UNIVERSITY OF  
MELBOURNE

# 1. Intro to Array

# How to store 5,00 students grades?



```
int s1_grade;  
int s2_grade;  
int s3_grade;  
int s4_grade;  
. . .  
int s50_grade;
```



```
int grade;  
for(int i = 0; i < 50; i++){  
    grade = ? ; // assign some value here  
}
```





# Array

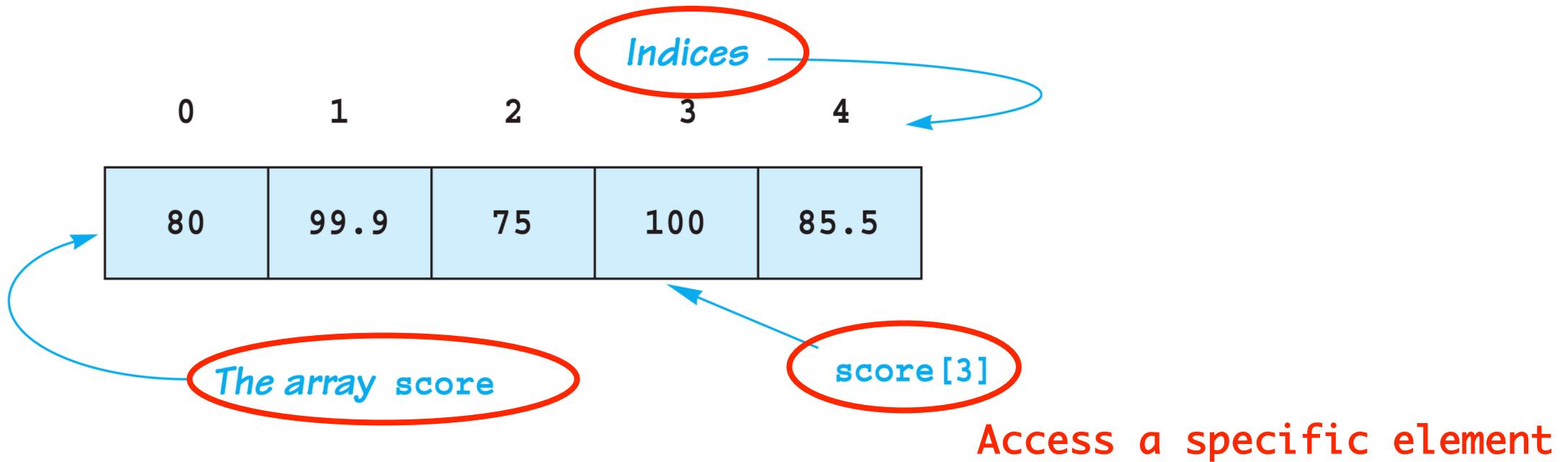


The goal of data structures is to organize and store information in an efficient manner.

- Basic data structure
- To store multiple values in a single variable, instead of declaring separate variables for each value.

# Create an array – option 1

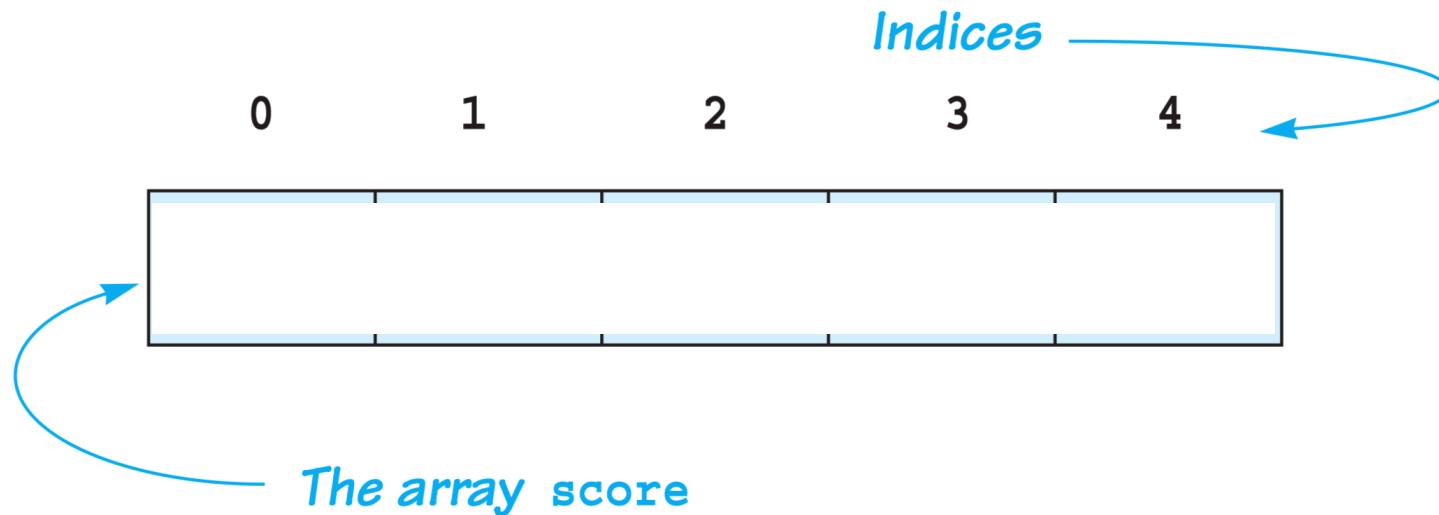
- `double[] score = {80, 99.9, 75, 100, 85.5};`





# Create an array – option 2

- `double[] score = new double[5];`





# Array properties

- `double[] score = {80, 99.9, 75, 100, 85.5};`
  - `double[] score = new double[5];`
- 
- size of the array has to be predefined
  - fixed size
  - store one type of object/primitive



# Change an array element



```
String[] pl = {"java", "python", "go", "javascript"};
```

```
pl[2] = "c++";
```



# Array Length VS Array size

```
int[] score = new int[9];
```



```
System.out.println(score.length());
```



# What happens if we try to access element outside the array length?



```
String[] pl = {"java", "python", "go", "javascript"};
```



```
System.out.println(pl[3]);
```



```
System.out.println(pl[4]);
```

JVM throws **ArrayIndexOutOfBoundsException**



THE UNIVERSITY OF  
MELBOURNE

## 2. Use an array



# Loop Through an Array

## Option 1 – loop over indices

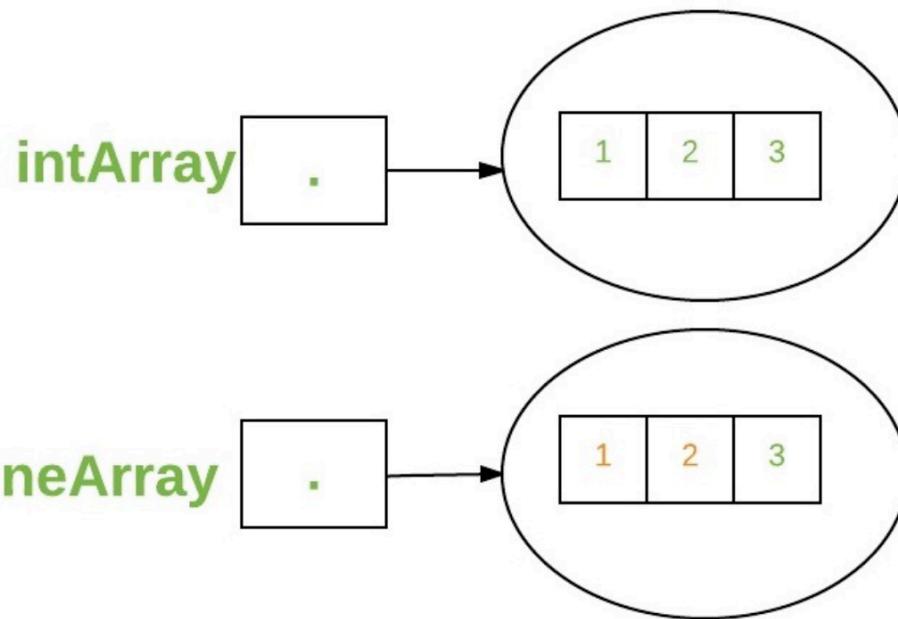
```
String[] cars = {"Volvo", "BMW", "Ford"};  
  
for (int i = 0; i < cars.length; i++) {  
    System.out.println(cars[i]);  
}
```

## Option 2 – for each loop

```
String[] cars = {"Volvo", "BMW", "Ford"};  
  
for (String car : cars) {  
    System.out.println(car);  
}
```



# Copy an Array



```
int[] original = {1,2,3};  
int[] cloneArray = original.clone();
```

False  
True

```
System.out.println(cloneArray == original);  
System.out.println(Arrays.equals(cloneArray, original));
```



THE UNIVERSITY OF  
MELBOURNE

### 3. Exercise



# Tutorial Exercise

1. Write a class **Subject** to represent a university subject. Each subject should always have a subject code, title, and coordinator, and should allow them to be changed. Also write a class **Student** to represent a university student. Each student should have a name, student number and username, which can be changed. Each student can be enrolled in up to four university subjects at once, and can be enrolled in and unenrolled from subjects. It should also be possible to find all the subjects a student is currently enrolled in.

Note that it is not a privacy leak for the **Student** class to return **Subject** objects or to store **Subject** objects passed to it, because those objects are not private. Consider the message enrolling a student in a subject. If some method later changes the name of that subject, then it is not an error for the subject recorded by that **Student** object to change; in fact, it would be an error for it not to change.

However, it *would* be a privacy leak for the **Student** class to give access to any array it may use to store subjects, because that *is* private. Allowing other classes to access that array would yield control over enrollment to other classes.



# Homework

2. Extend your **Subject** class so that it is possible to find all the students enrolled in a subject. You may assume that no more than 1000 students are enrolled in any subject at a time.
3. How could you implement the **Subject** class so that there is *no* limit on the number of students enrolled in a subject?



# Thank you





**WARNING**

This material has been reproduced and communicated to you by or on behalf of the University of Melbourne in accordance with section 113P of the *Copyright Act 1968 (Act)*.

The material in this communication may be subject to copyright under the Act.

Any further reproduction or communication of this material by you may be the subject of copyright protection under the Act.

**Do not remove this notice**