

MODULE 5, 6

Module 5:

Arrays/Strings/ Other string classes

Module 6:

Packages , Access Modifiers

Review

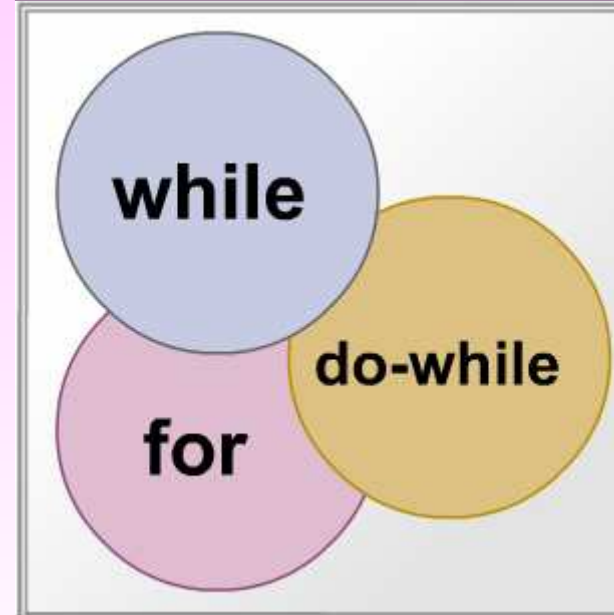
Decision-making statements

- ✓ if statement
- ✓ if-else-if statement
- ✓ switch-case statement

Jump statements

break;
continue;

Loop statements



Review

- **Class** = data + methods
- **Object** = an instance of a class
- **Instance variable**: a variable contains a reference to an object.
- **Method**: a behavior of a class (code)
- **Constructor**: a method which is executed when an object is created.
- **Initializer**: Codes which will be executed just after an object is created to set the initial values to data of object.

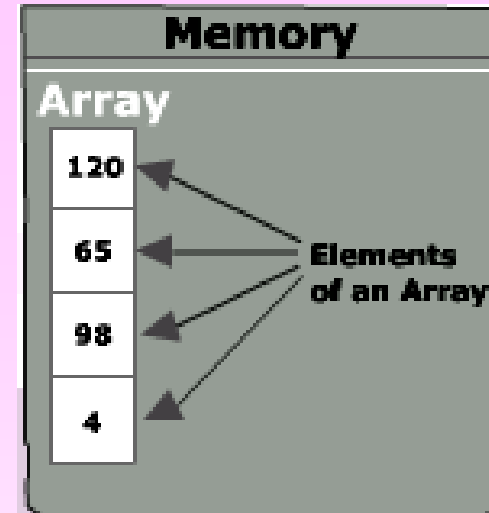
Module 5:

- Arrays
- String
- StringBuilder
- StringTokenizer

Module 5: Arrays

Definition

An array is a special data store that can hold several items of a single data type in contiguous memory locations.



Benefits of using Arrays



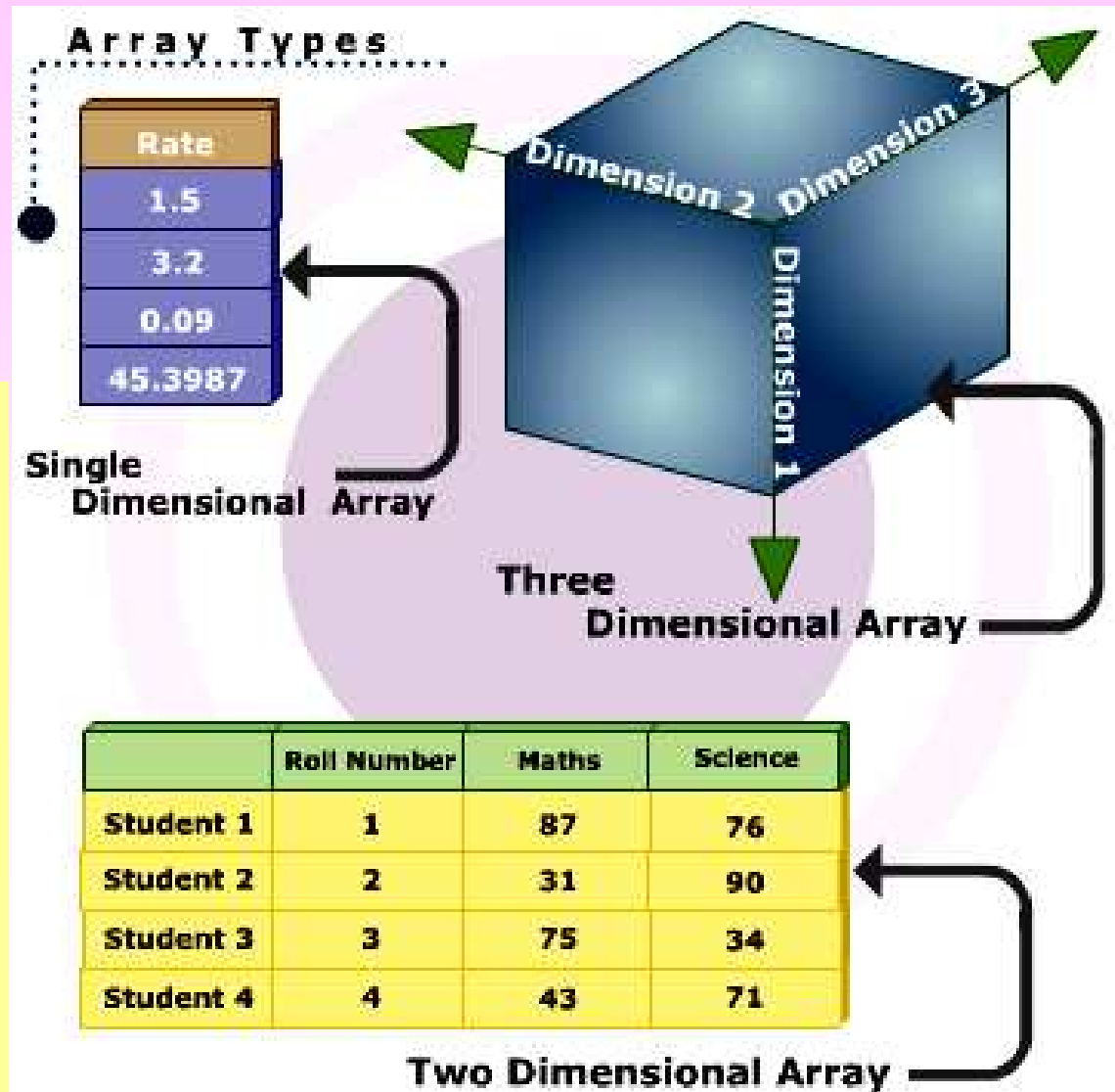
- Arrays are the best means of operating on multiple data elements of the same type at the same time.
- Arrays make optimum use of memory resource as compared to variables.
- Memory can be assigned to an array only at the time when the array is actually used. Thus, the memory is not consumed by an array right from the time you declare it.

Types of arrays

```
for (int i=0; i<ar.length; i++)  
// for (Type var : ar)// JDK 1.5  
{ // if (condition)  
  process a[i];  
}
```

Type [] ar;

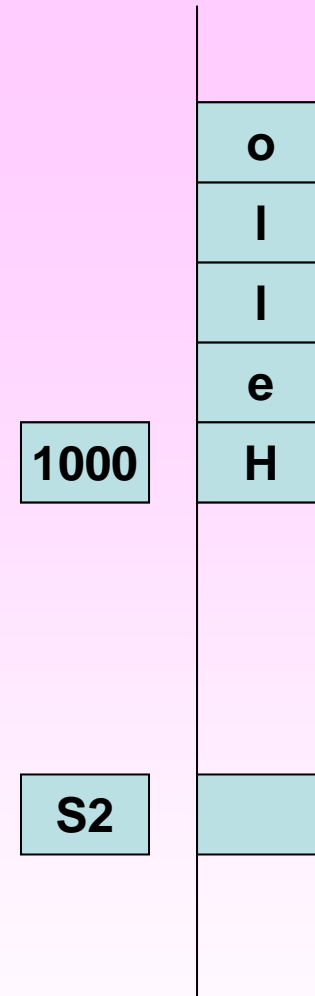
- Declaring:
Type[][] m; // or Type m[][];
- Allocating memory:
m = new Type[RowNo][ColNo];
- Element accessing:
m[r_index][c_index] // index >= 0
- Traversing a 2D array (Type[][]m)
for (int i=0; i<m.length;i++)
{ for (int j=0; j<m[i].length;j++)
{ // if (condition)
 process m[i][j];
}
}



String

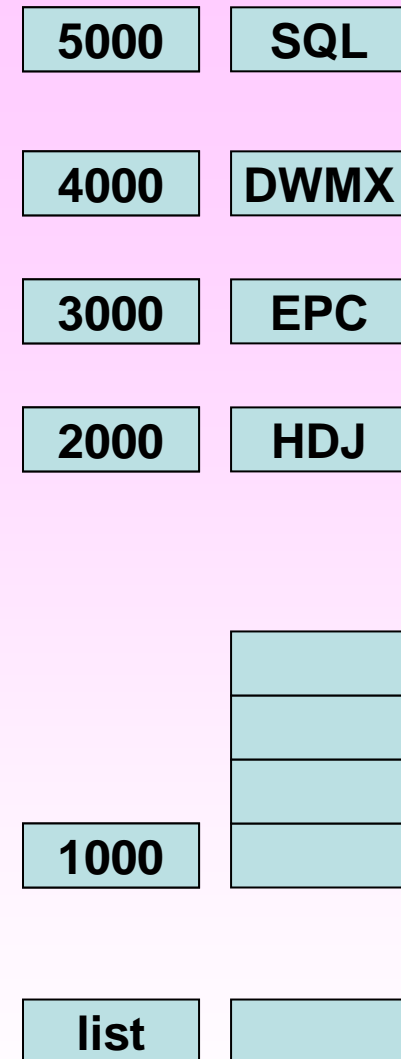
- Class for immutable text data.
- **Declare:** `String S1 = new String();`
`String S2= "Hello";`
- Common-used methods

- | | |
|-------------|---------------|
| length() | lastIndexOf() |
| charAt() | replace() |
| concat() | substring() |
| compareTo() | toString() |
| indexOf() | trim() |



String Arrays

- `String[] list = { "EPC", "HDJ",
"DWMX", "SQL",};`
`for (int i=0;i< list.length;i++)`
`System.out.println(list[i]);`



Passing command line arguments

```
class A
{ .....
  public static void main (String[] args)
  { // manipulate with args[i]
  }
}
// run program
java A arg0 arg1 arg2
```

StringBuilder class

Store a growable and flexible string.

Constructors

Common used methods

- append()
- insert()
- delete()
- reverse()

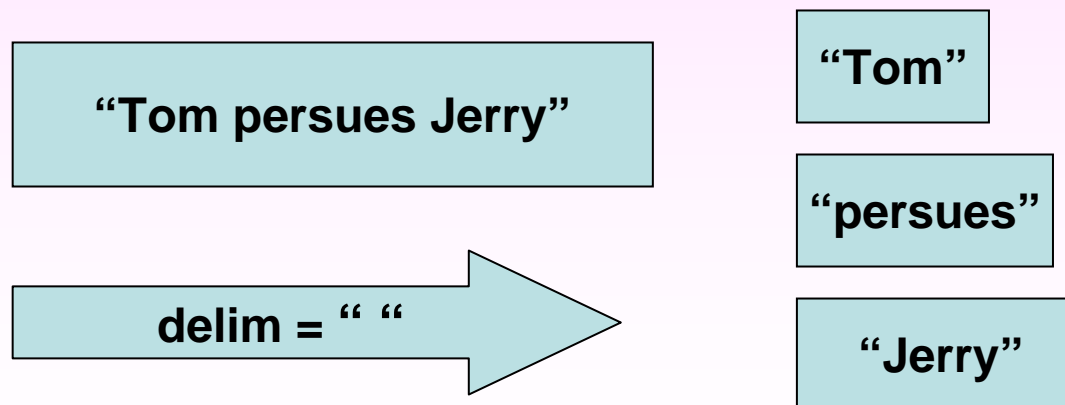
- `StringBuilder()` : default constructor reserves room for 16 characters.
- `StringBuilder(int capacity)` : constructs an object with no characters in it but reserves room for characters specified in the capacity argument.
- `StringBuilder(String str)` : constructs an object that is initialized to the contents of the specified string, `str`.

StringTokenizer class

- Class for break a string into subunits

```
countTokens()  
hasMoreElements()  
hasMoreTokens()  
nextElement()  
nextToken()
```

```
StringTokenizer(String str)  
StringTokenizer(String str, String delim)  
StringTokenizer(String str, String delim, boolean returnDelims)
```



Module 6:

- Packages
- Access Modifiers

Packages

Packages

A group of related Java classes and interfaces organized as a unit in namespaces is termed a package.

Types of Java packages are:

- Predefined packages
- User-defined packages

Predefined Packages



You can view it's content with WINZAR/ WINZIP

```
import java.util.*;  
import java.io.*;  
class A  
{ .....  
}
```

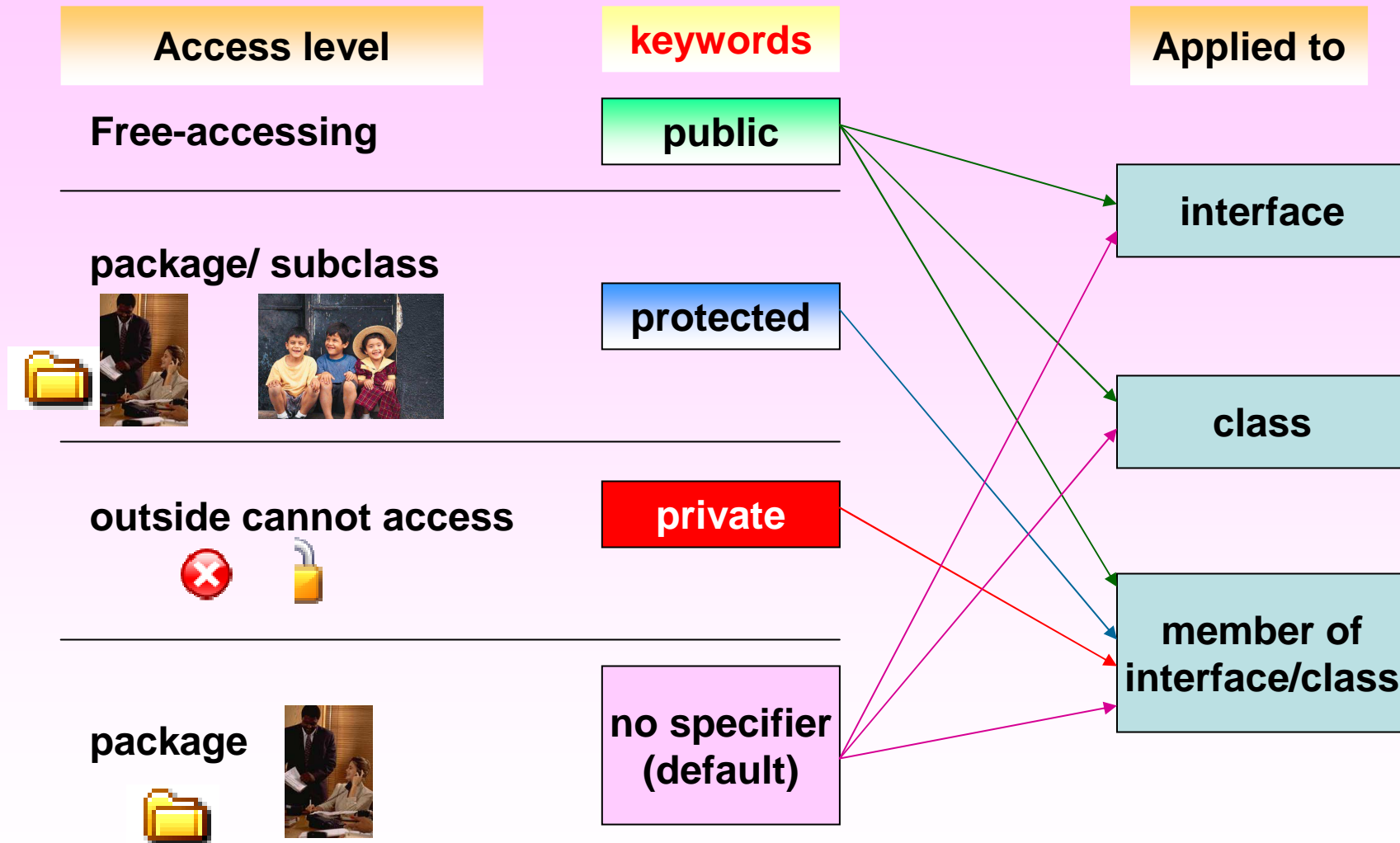
User-defined Packages

- Create a package
`package my_package;`
`class A`
`{`

`}`

- Use the package
`import my_package.A;`
`class B`
`{`
`void method3()`
`{ A obj= new A();`
`.....`
`}`
`}`

Access control keyword



Access control keywords...

Access modifier	Elements Visible in			
	Class	Package	Subclass (in other package)	Outside
public	✓ yes	✓ yes	✓ yes	✓ yes
protected	✓ yes	✓ yes	✓ yes	✗ no
private	✓ yes	✓ yes	✗ no	✗ no
no modifier	✓ yes	✓ yes	✗ no	✗ no

Field and method Modifiers

Keywords for specifying changing permission

- volatile (dễ biến đổi) modifier
- final (cuối cùng) modifier
- native (cục bộ) modifier
- transient (tạm thời) modifier

Summary

- Java package is a group of related classes and interfaces organized as one unit.
- Packages can be classified as build-in or predefined packages and user-defined packages.
- The package statement, if it is added, must be the first line in Java code.
- The import statement is used to import one or more classes from a package into a program.
- Access modifiers control the access and visibility of a class and class members.
- Field and method modifiers are keywords used to identify fields or methods that need to be declared for controlling access to users.

Thanks