

Java Programming AP Edition

U1C2 Elementary Programming

IDENTIFIERS

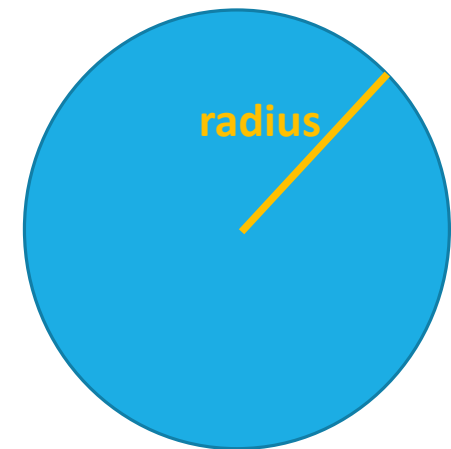
ERIC Y. CHOU, PH.D.

IEEE SENIOR MEMBER



Where will see a lot of identifiers

```
public class Example {  
    public static void main(String[] args){  
        // Variable Declaration  
        double radius = 5.0;  
        // input part  
        Scanner input = new Scanner(System.in);  
        radius = input.nextDouble();  
        // Processing part  
        double area = Math.PI * radius * radius;  
        // Output Part  
        System.out.println(area);  
    }  
}
```





Identifiers

Identifiers are the name that identify the elements such as **variables, classes, methods** in a program.

- All identifiers in Java must obey the following rules:
- An identifier is a sequence of characters that consists of **letters, digits underscores (_), and dollar sign (\$)**. An identifier must start with **a letter, an underscore (_), or a dollar sign (\$)**. It can not start with a digit (number).
- An identifier cannot be a reserved word.
- An identifier cannot be **true, false, or null**.
- An identifier can be of any length.



Variables

Variables are used to represent values that may be changed in the program.

A variable must be declared before used.

A variable declaration in syntax: `<data type> <variable name>;`

Examples of variable declarations:

❖ `int count;`

❖ `double radius;`

❖ `double interestRate;`

Primitive data type: `byte, short, int, long, float, double, char, Boolean;`

Reference data type: `<class name>`

Multiple instance for variable declaration is allowed in Java:

`datatype variable1, variable2, ..., variablen;`

Declaration and assignment in the same statement is also allowed: `int count=1;`



Declaring Variables

```
int x;           // Declare x to be an
                 // integer variable;

double radius; // Declare radius to
               // be a double variable;

char a;         // Declare a to be a
               // character variable;
```



Assignment Statements

```
x = 1;           // Assign 1 to x;
```

```
radius = 1.0;   // Assign 1.0 to radius;
```

```
a = 'A';        // Assign 'A' to a;
```



Variables

```
// Compute the first area
```

```
radius = 1.0;
```

```
area = radius * radius * 3.14159;
```

```
System.out.println("The area is " + area + " for radius "+radius);
```

```
// Compute the second area
```

```
radius = 2.0;
```

```
area = radius * radius * 3.14159;
```

```
System.out.println("The area is " + area + " for radius "+radius);
```



Assignment Statement and Assignment Expressions

An assignment statement designates a value for a variable. An assignment statement can be used as an expression in Java.

Syntax (assignment statement):

```
<variable> = <expression> ;
```

An expression represents a computation involving values, variables, and operators that taking them together, evaluates to a value.

```
int y=1;  
double radius = 1.0;  
int x = 5 * (3 / 2) ;
```

```
x = 1; // correct;  
1 = x; // incorrect;
```




Assignements

Evaluate the + sign first, then the assignment.

```
x = 1;
```

```
x = x + 1;
```

If a value is assigned to multiple variables, you can use this syntax:

```
i = j = k = 1;
```

Which is equivalent to:

```
k = 1;  
j = k;  
i = j;
```



Identifiers for Methods

```
public static void main(String[] args){ ... }
```

```
public double abs(double a){ ... }
```

Will be covered in Chapter 9.