Introduction to computer programming A
LAB2

贾艳红 Jana
Email:jiayh@mail.sustech.edu.cn

LAB OBJECTIVES

1 Learn how to use an Integrated Development Environment (IDE) in writing JAVA programs

2 Primitive types

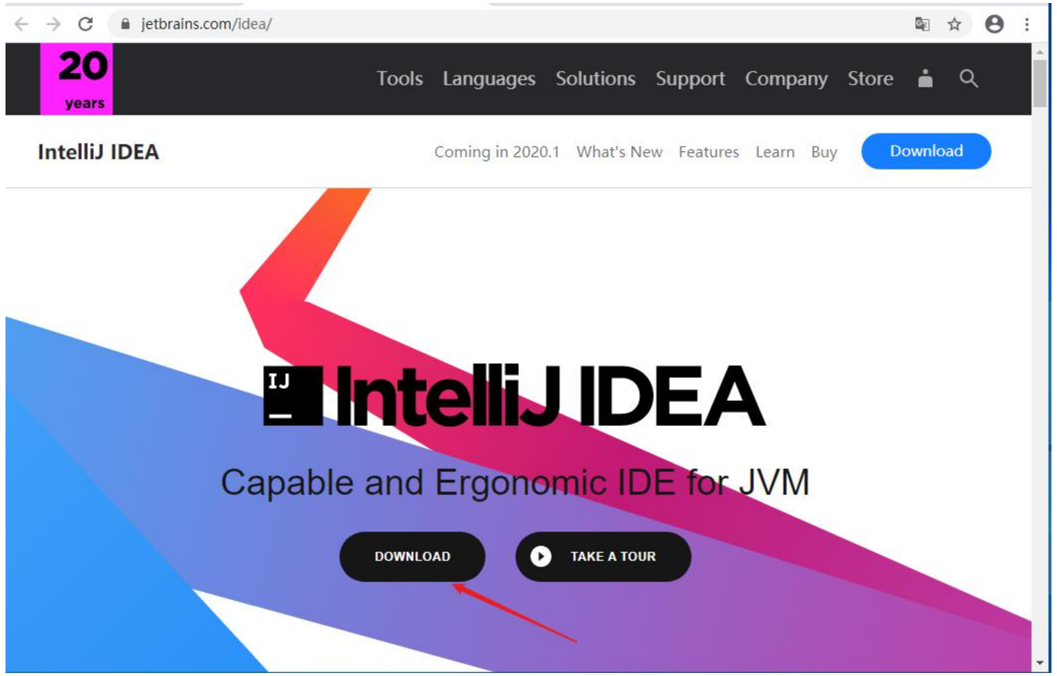
3 Practice using input and output statements.

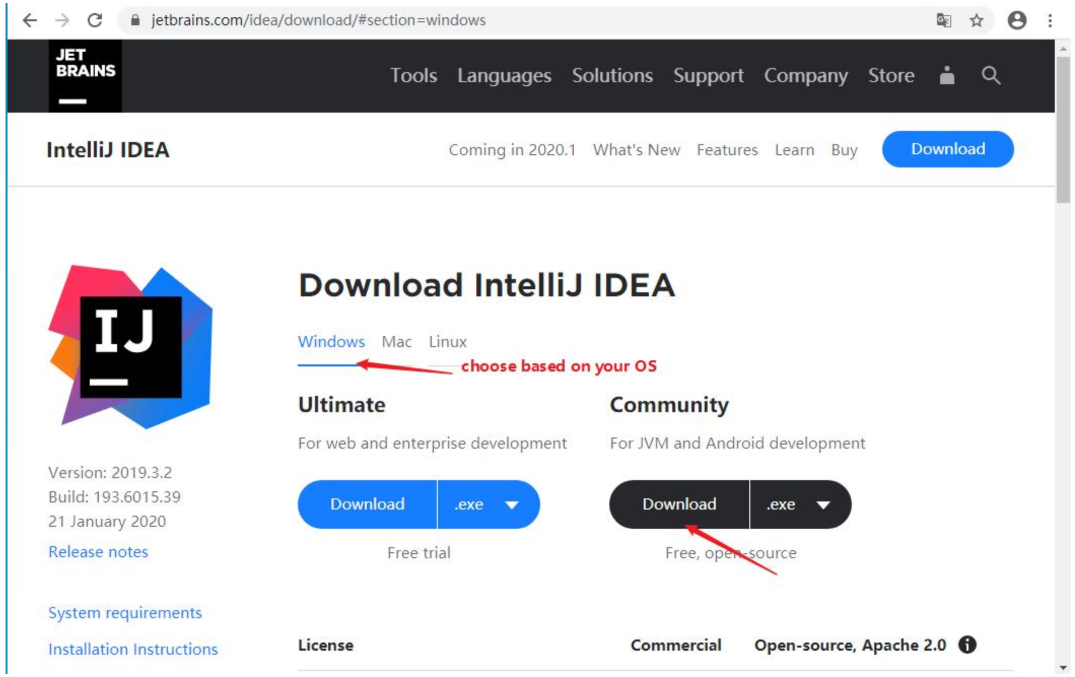
4 Selections

1 DownLoading and Installing IDEA

DownLoading IDEA

Download IDEA Installer from:https://www.jetbrains.com/idea/

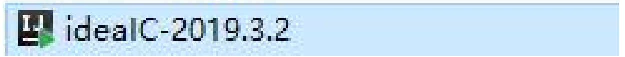

jetbrains.com/idea/
20 years
Tools Languages Solutions Support Company Store
IntelliJ IDEA
Coming in 2020.1 What's New Features Learn Buy Download
IJ IntelliJ IDEA
Capable and Ergonomic IDE for JVM
DOWNLOAD
TAKE A TOUR


jetbrains.com/idea/download/#section=windows
JET BRAINS
Tools Languages Solutions Support Company Store
IntelliJ IDEA
Coming in 2020.1 What's New Features Learn Buy Download
IJ
Download IntelliJ IDEA
Windows Mac Linux
Ultimate
For web and enterprise development
Download .exe
Free trial
Community
For JVM and Android development
Download .exe
Free, open-source
Release notes
System requirements
Installation Instructions
License
Commercial
Open-source, Apache 2.0

choose based on your OS

Version: 2019.3.2
Build: 193.6015.39
21 January 2020

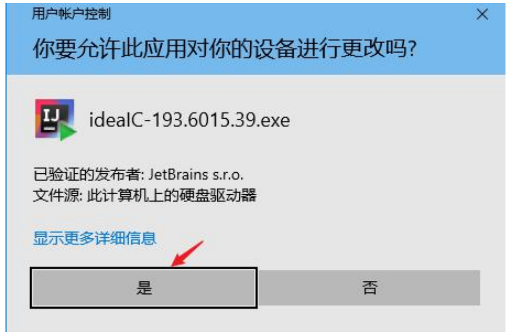
The idea installation package downloaded looks like this:



ideaIC-2019.3.2

Installing IDEA

- Once the download is complete, locate the ***ideaIC-2019.3.2.exe*** file and **double-click to run the installer**.

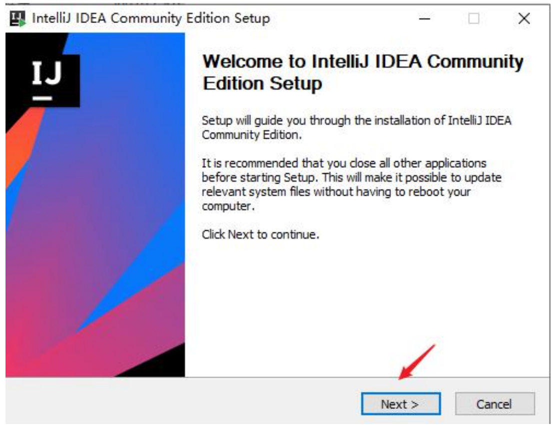


用户帐户控制

你要允许此应用对你的设备进行更改吗?

ideaIC-193.6015.39.exe

已验证的发布者: JetBrains s.r.o.
文件源: 此计算机上的硬盘驱动器



IntelliJ IDEA Community Edition Setup

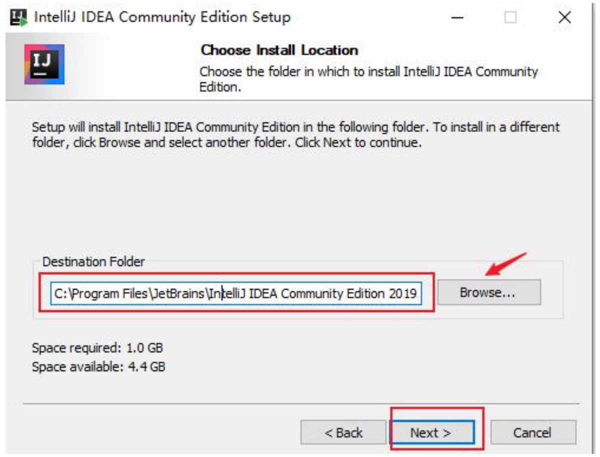
Welcome to IntelliJ IDEA Community Edition Setup

Setup will guide you through the installation of IntelliJ IDEA Community Edition.

It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.

- Click ***Next*** and on the following screen optionally change the installation location by clicking on the ***Change...*** button. In this example the default install location was kept.



IntelliJ IDEA Community Edition Setup

Choose Install Location

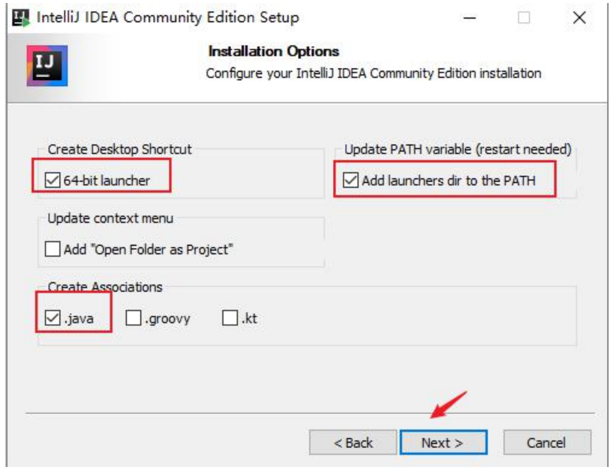
Choose the folder in which to install IntelliJ IDEA Community Edition.

Setup will install IntelliJ IDEA Community Edition in the following folder. To install in a different folder, click Browse and select another folder. Click Next to continue.

Destination Folder

C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2019

Space required: 1.0 GB
Space available: 4.4 GB



IntelliJ IDEA Community Edition Setup

Installation Options

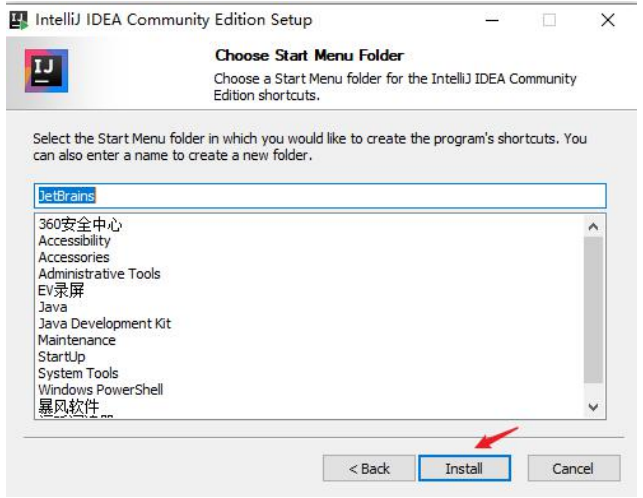
Configure your IntelliJ IDEA Community Edition installation

Create Desktop Shortcut
☑ 64-bit launcher

Update PATH variable (restart needed)
☑ Add launchers dir to the PATH

Update context menu
□ Add "Open Folder as Project"

Create Associations
☑ .java □ .groovy □ .kt



IntelliJ IDEA Community Edition Setup

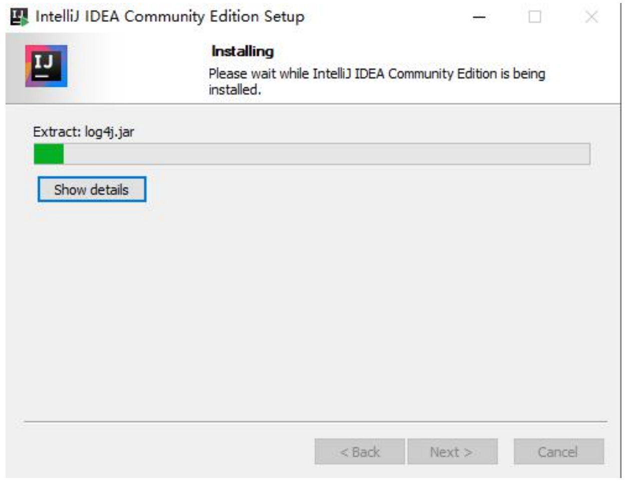
Choose Start Menu Folder

Choose a Start Menu folder for the IntelliJ IDEA Community Edition shortcuts.

Select the Start Menu folder in which you would like to create the program's shortcuts. You can also enter a name to create a new folder.

JetBrains

360安全中心
Accessibility
Accessories
Administrative Tools
EV录屏
Java
Java Development Kit
Maintenance
StartUp
System Tools
Windows PowerShell
暴风软件



IntelliJ IDEA Community Edition Setup

Installing

Please wait while IntelliJ IDEA Community Edition is being installed.

Extract: log4j.jar

Create a new Java project

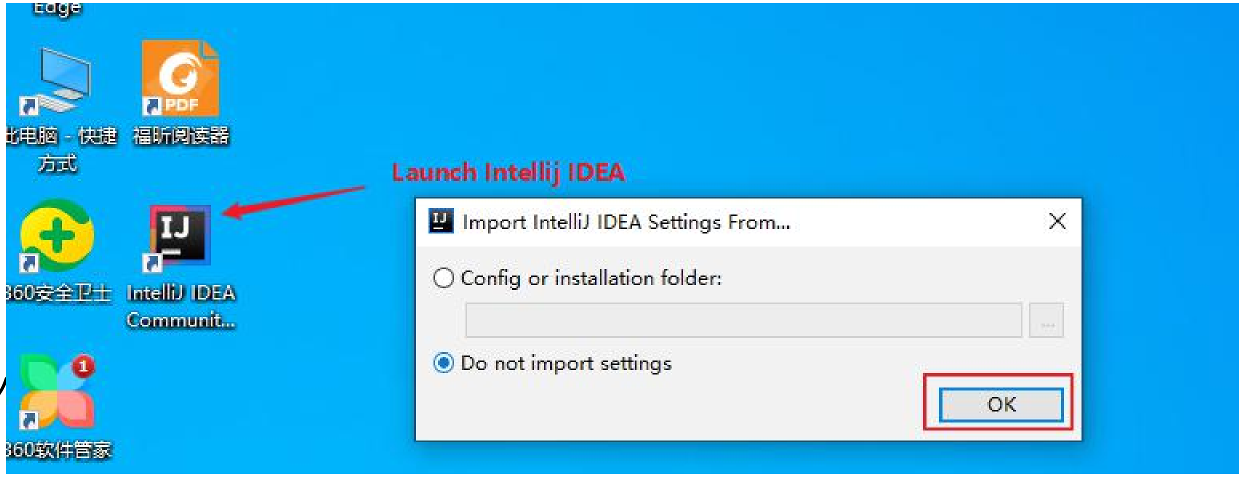
1. Launch IntelliJ IDEA.

*If the Welcome screen opens, click **Create New Project**.Otherwise, from the main menu, select **File | New | Project.***

2.In the **New Project** wizard, select Java from the list on the left.

3.From the **Project SDK** list, select the JDK that you want to use in your project.

*If the list is empty, click New and specify the path to the Java home directory (for example,**C:\Program Files\Java\jdk1.8.0_212**).*



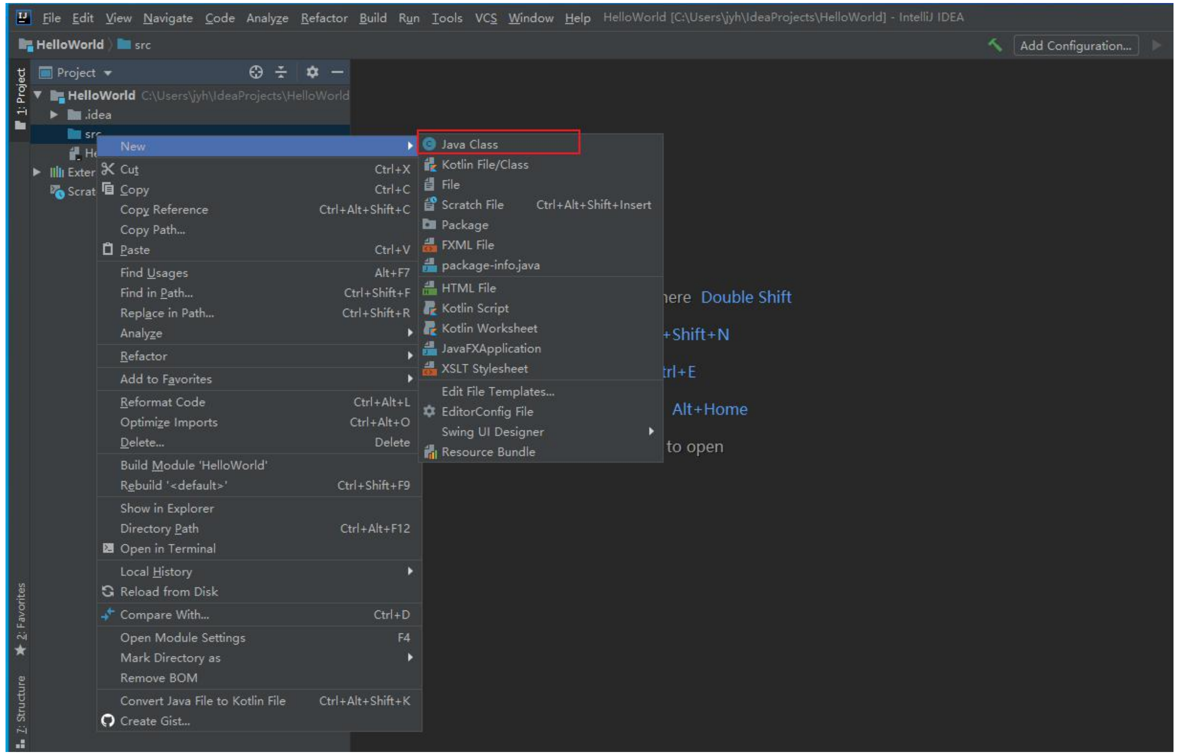
Launch Intellij IDEA

Import IntelliJ IDEA Settings From...

○ Config or installation folder:

● Do not import settings

OK

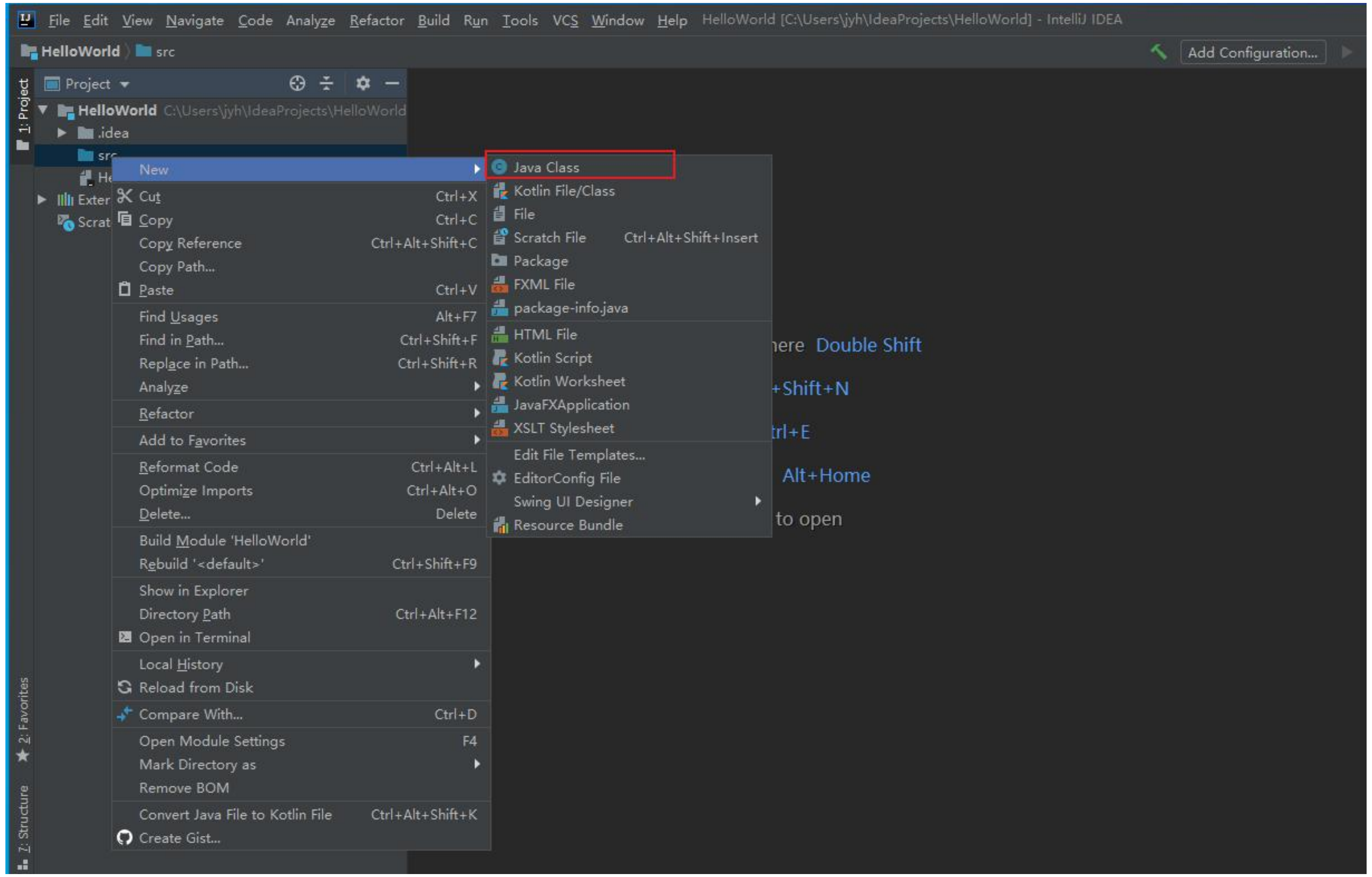


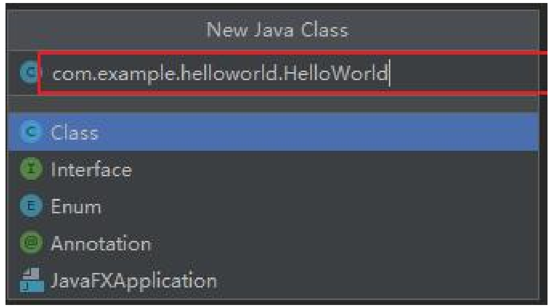
Create a package and a class

1. In the **Project** tool window, select the **src** folder, right-click on it, and select **Java Class**.

2.In the Name field, type ***com.example.helloworld.***HelloWorld and **click** OK.

*IntelliJ IDEA creates the **com.example.helloworld** package and the **HelloWorld** class.*





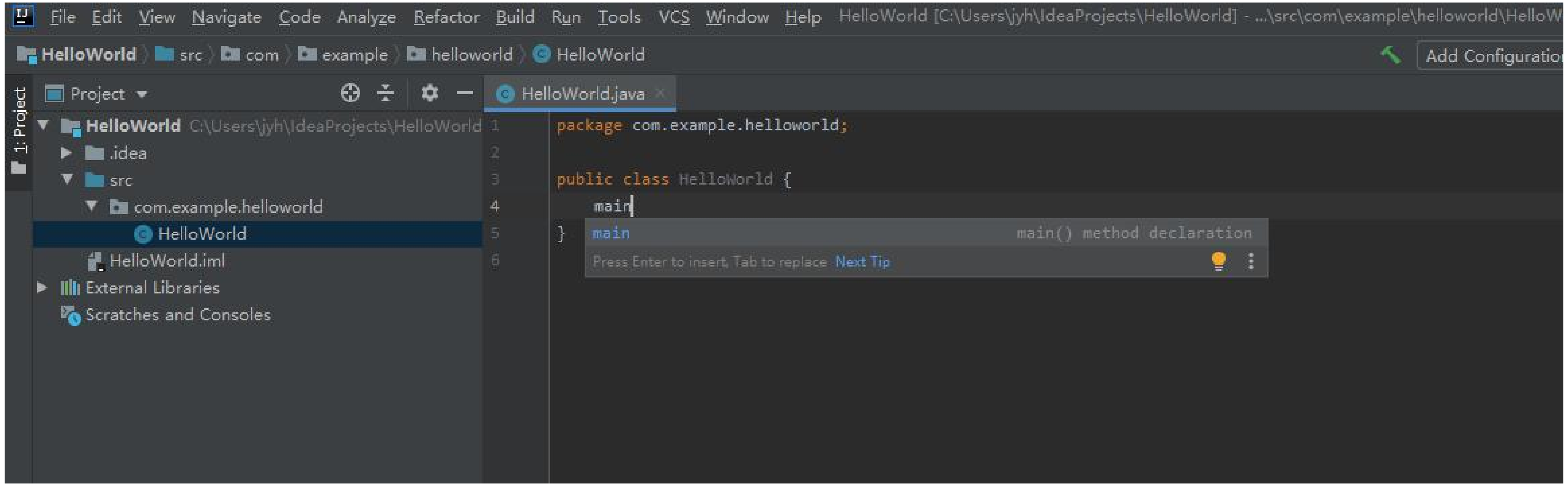
com.example.helloworld.HelloWorld

Note:
In Java, there's a **naming convention** that you should follow when you name packages and classes.

https://www.oracle.com/technetwork/java/codeconventions-135099.html

Write the code

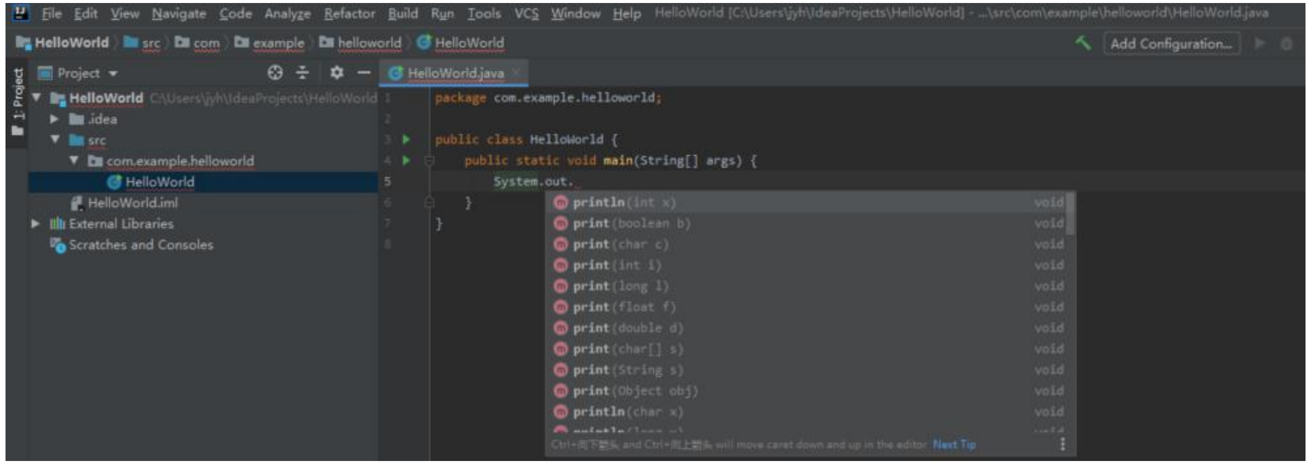
1.Type **main** and select the template that inserts the **main**() method declaration.


File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help HelloWorld [C:\Users\jyh\IdeaProjects\HelloWorld] - ...\src\com\example\helloworld\HelloW
HelloWorld src com example helloworld HelloWorld
Add Configuratio
1: Project
Project
HelloWorld.java
HelloWorld C:\Users\jyh\IdeaProjects\HelloWorld
.idea
src
com.example.helloworld
HelloWorld
HelloWorld.iml
External Libraries
Scratches and Consoles
main
main() method declaration

```java
package com.example.helloworld;

public class HelloWorld {
    main
}
```

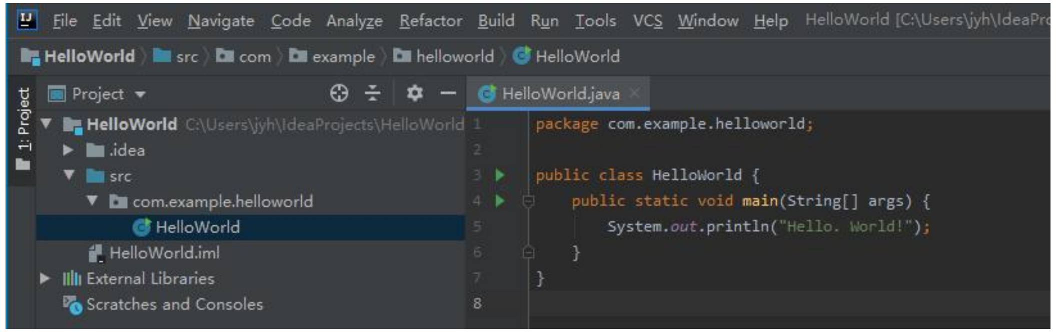
Press Enter to insert, Tab to replace Next Tip

2.Call the **println**() method using code completion



```java
package com.example.helloworld;

public class HelloWorld {
    public static void main(String[] args) {
        System.out.
    }
}
```

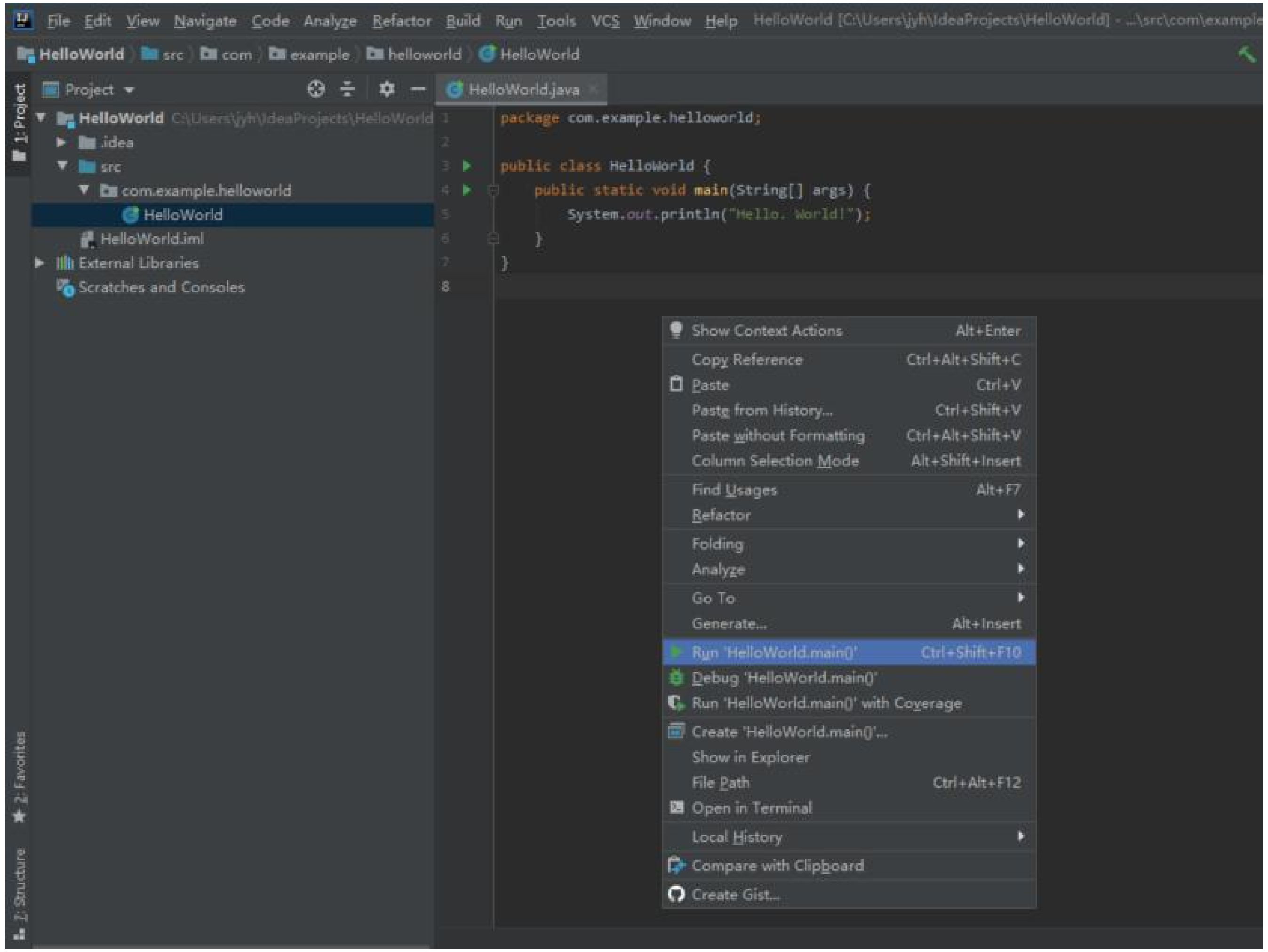
Ctrl+向下箭头 and Ctrl+向上箭头 will move caret down and up in the editor Next Tip


File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help HelloWorld [C:\Users\jyh\IdeaPro
HelloWorld src com example helloworld HelloWorld
1: Project
Project
HelloWorld.java
HelloWorld C:\Users\jyh\IdeaProjects\HelloWorld
.idea
src
com.example.helloworld
HelloWorld
HelloWorld.iml
External Libraries
Scratches and Consoles

```java
package com.example.helloworld;

public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello. World!");
    }
}
```

Build and run the application

1.Go to the Run menu and select the Run option.
2. Select the Class name and click on Run.
3.right-click editor and elect Run 'HelloWorld.main()' in the popup

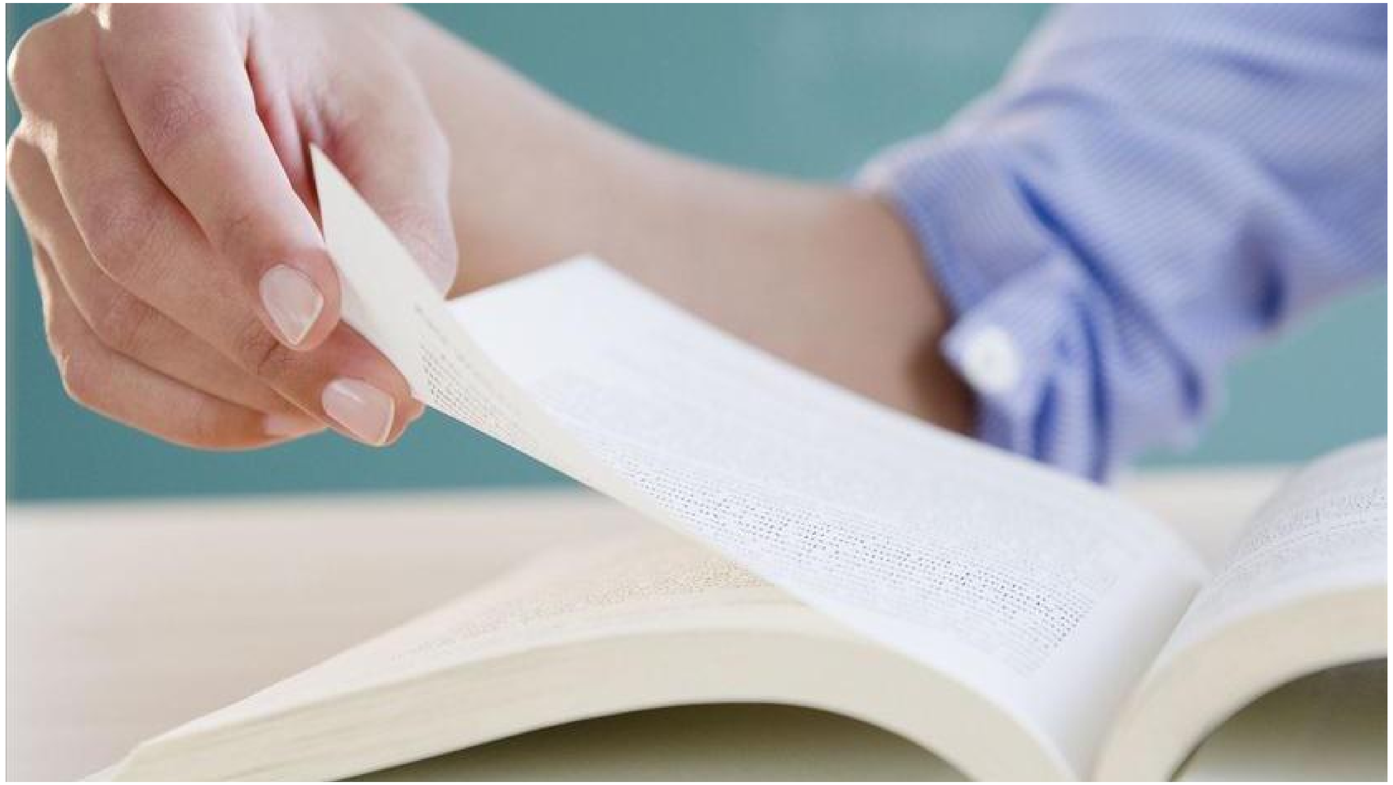


File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help HelloWorld [C:\Users\jyh\IdeaProjects\HelloWorld] - ...\src\com\example
HelloWorld src com example helloworld HelloWorld
Project
HelloWorld.java
HelloWorld C:\Users\jyh\IdeaProjects\HelloWorld
.idea
src
com.example.helloworld
HelloWorld
HelloWorld.iml
External Libraries
Scratches and Consoles
1: Project
2: Favorites
7: Structure

```
package com.example.helloworld;

public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello, World!");
    }
}
```

Show Context Actions Alt+Enter
Copy Reference Ctrl+Alt+Shift+C
Paste Ctrl+V
Paste from History... Ctrl+Shift+V
Paste without Formatting Ctrl+Alt+Shift+V
Column Selection Mode Alt+Shift+Insert
Find Usages Alt+F7
Refactor
Folding
Analyze
Go To
Generate... Alt+Insert
Run 'HelloWorld.main()' Ctrl+Shift+F10
Debug 'HelloWorld.main()'
Run 'HelloWorld.main()' with Coverage
Create 'HelloWorld.main()'...
Show in Explorer
File Path Ctrl+Alt+F12
Open in Terminal
Local History
Compare with Clipboard
Create Gist...

2 Primitive types



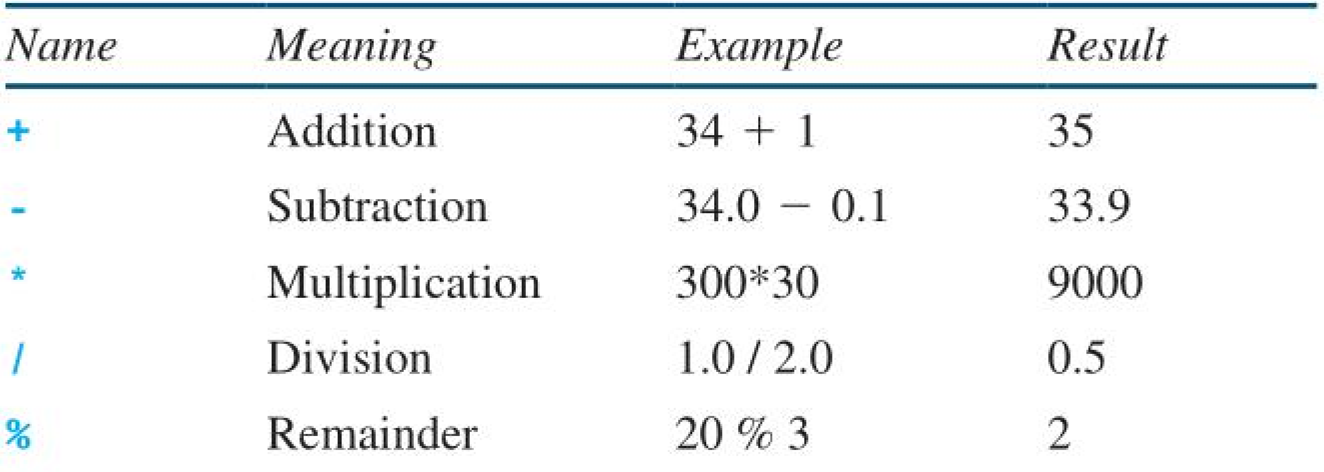
Data type and its range

Table:List of Java's primitive data types

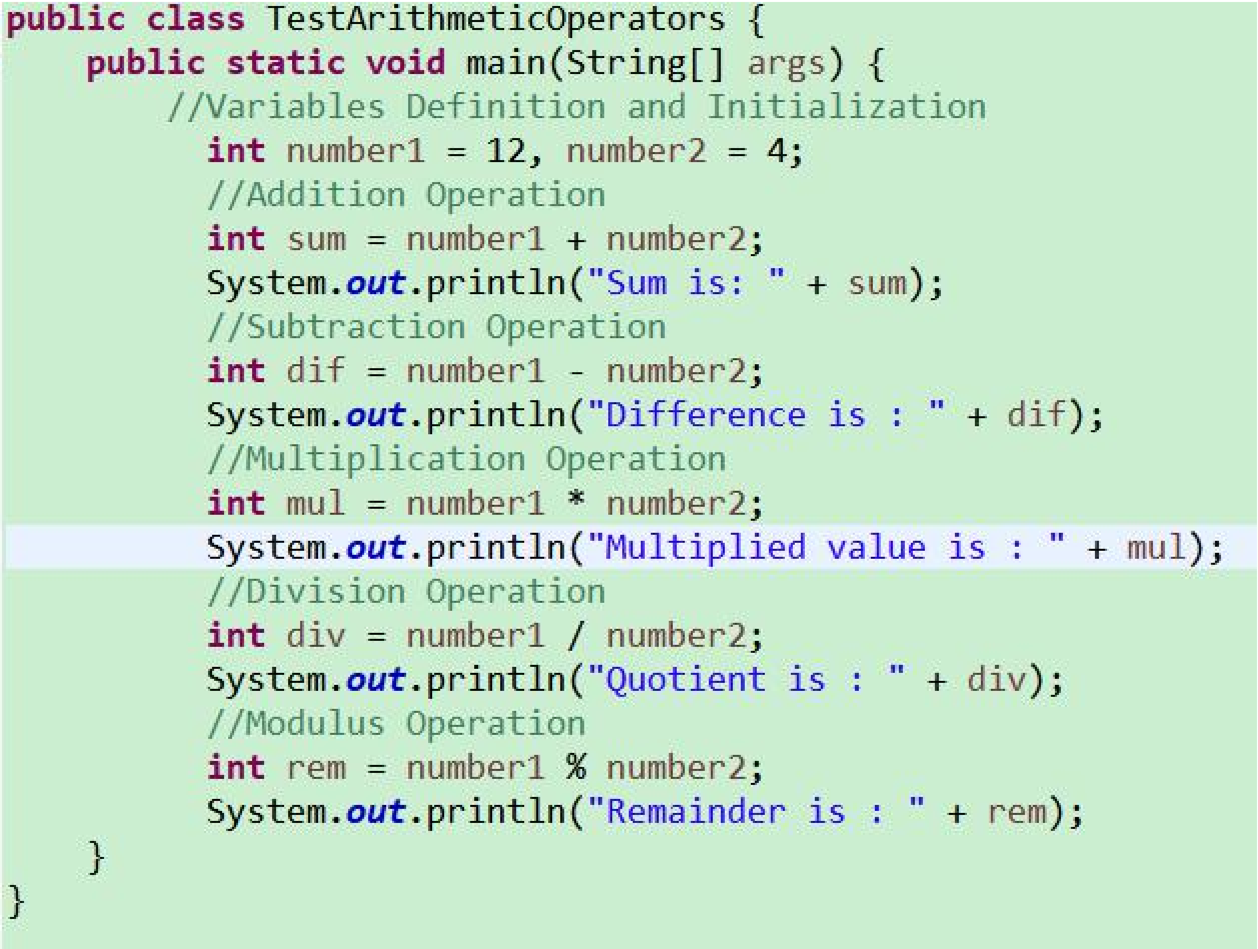
Type	Size in Bytes	Range
byte	1 byte	-2^7 to 2^7 -1 (-128 to 127)
short	2 bytes	-2^{15} to 2^{15} -1 (-32768 to 32767)
int	4 bytes	-2^{31} to 2^{31} -1 (-2147483648 to 2147483647)
long	8 bytes	-2^{63} to 2^{63} -1(-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807)
float	4 bytes	approximately ±3.40282347E+38F (6-7 significant decimal digits) Java implements IEEE 754 standard
double	8 bytes	approximately ±1.79769313486231570E+308 (15 significant decimal digits)
char	2 byte	0 to 65,536 (unsigned)
boolean	not precisely defined*	true or false

*boolean represents one bit of information, but its "size" isn't something that's precisely defined

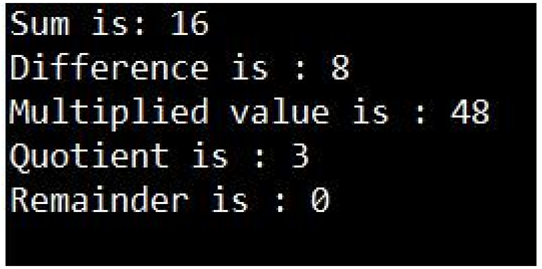
Data Operations(Basic Arithmetic Operators)



Name	*Meaning*	*Example*	*Result*
+	Addition	34 + 1	35
-	Subtraction	34.0 – 0.1	33.9
*	Multiplication	300*30	9000
/	Division	1.0 / 2.0	0.5
%	Remainder	20 % 3	2



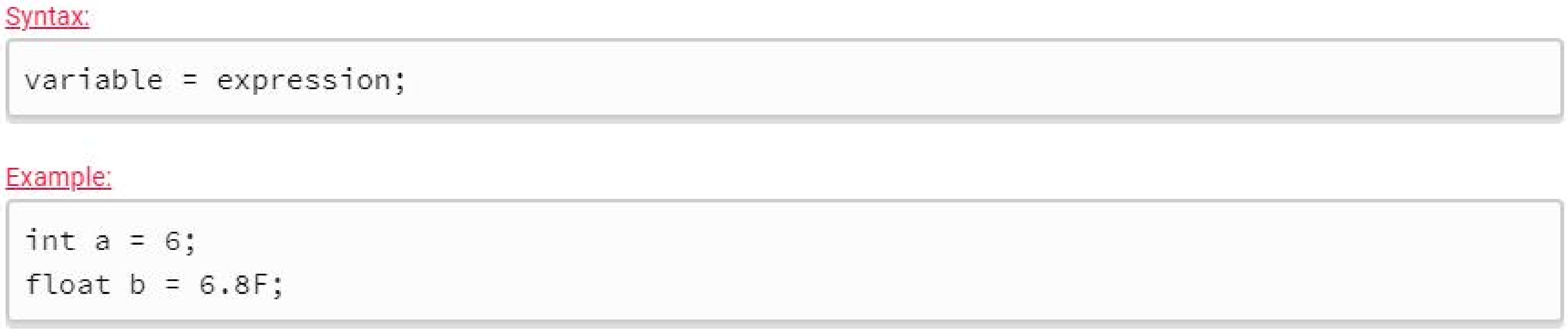
```java
public class TestArithmeticOperators {
    public static void main(String[] args) {
        //Variables Definition and Initialization
          int number1 = 12, number2 = 4;
          //Addition Operation
          int sum = number1 + number2;
          System.out.println("Sum is: " + sum);
          //Subtraction Operation
          int dif = number1 - number2;
          System.out.println("Difference is : " + dif);
          //Multiplication Operation
          int mul = number1 * number2;
          System.out.println("Multiplied value is : " + mul);
          //Division Operation
          int div = number1 / number2;
          System.out.println("Quotient is : " + div);
          //Modulus Operation
          int rem = number1 % number2;
          System.out.println("Remainder is : " + rem);
    }
}
```



```
Sum is: 16
Difference is : 8
Multiplied value is : 48
Quotient is : 3
Remainder is : 0
```

Data Operations(Assignment Operators)

The Java Assignment Operators are used when you want to assign a value to the expression. The assignment operator denoted by the single equal sign =.



Syntax:

```
variable = expression;
```

Example:

```
int a = 6;
float b = 6.8F;
```

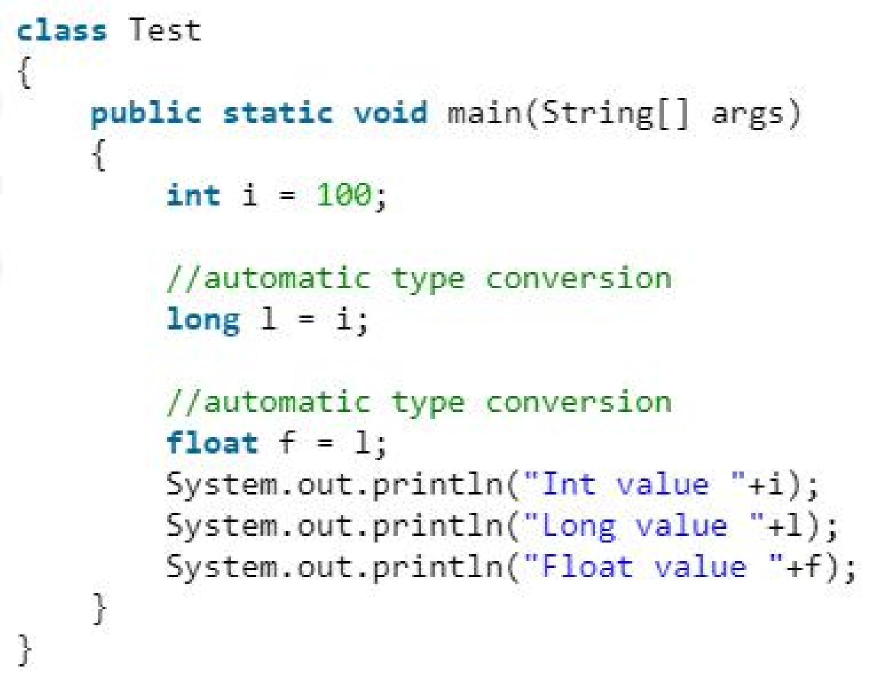
Data type Conversions

◆Widening or Automatic Type Conversion

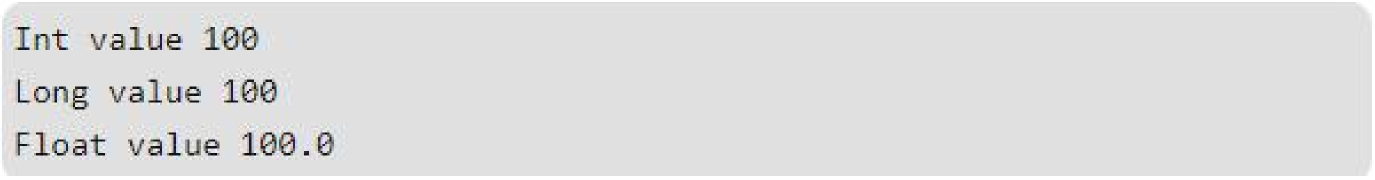
- The two data types are compatible.(char and boolean are not compatible with each other.)
- When we assign value of a smaller data type to a bigger data type.

Byte –> Short –> Int –> Long – > Float –> Double

Example:



```
class Test
{
    public static void main(String[] args)
    {
        int i = 100;

        //automatic type conversion
        long l = i;

        //automatic type conversion
        float f = l;
        System.out.println("Int value "+i);
        System.out.println("Long value "+l);
        System.out.println("Float value "+f);
    }
}
```

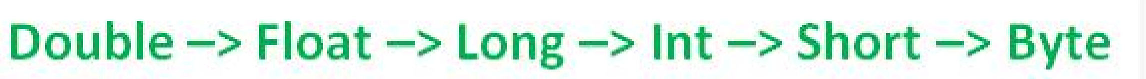


```
Int value 100
Long value 100
Float value 100.0
```

Data type Conversions

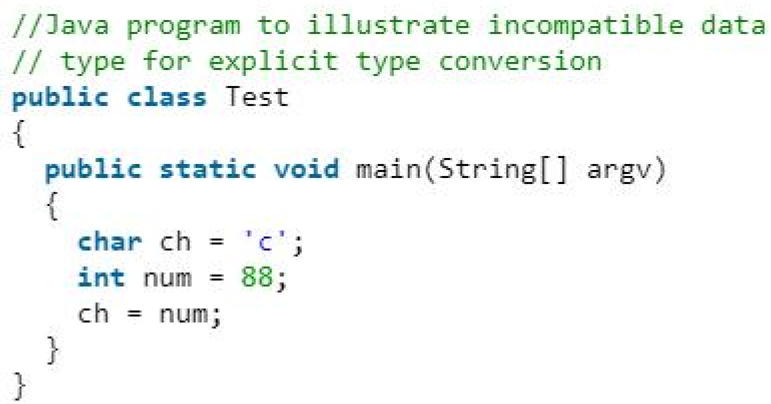
◆Narrowing or Explicit Conversion

- This is useful for incompatible data types where automatic conversion cannot be done.
- Assign a value of larger data type to a smaller data type

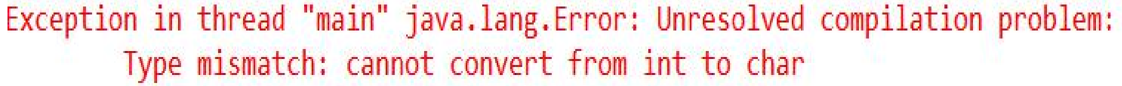


Double –> Float –> Long –> Int –> Short –> Byte

Example:

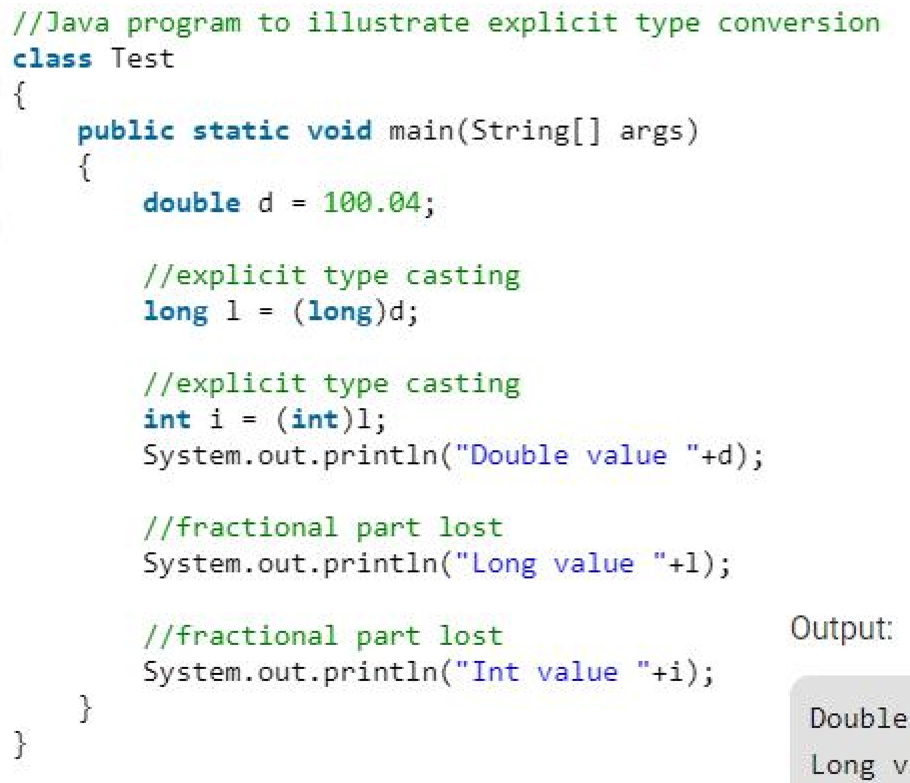


```
//Java program to illustrate incompatible data
// type for explicit type conversion
public class Test
{
  public static void main(String[] argv)
  {
    char ch = 'c';
    int num = 88;
    ch = num;
  }
}
```



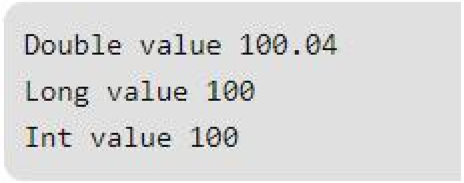
```
Exception in thread "main" java.lang.Error: Unresolved compilation problem:
	Type mismatch: cannot convert from int to char
```

Example:



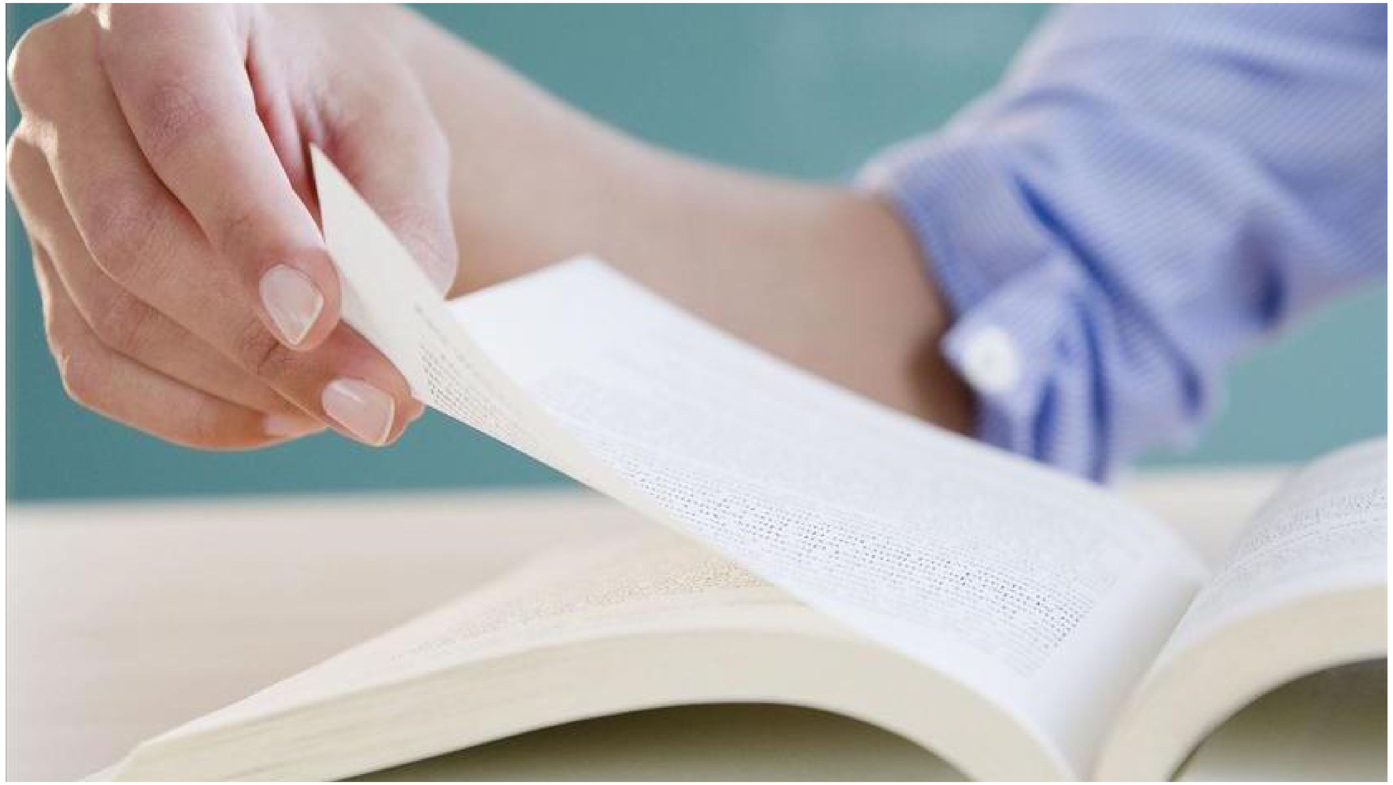
```
//Java program to illustrate explicit type conversion
class Test
{
    public static void main(String[] args)
    {
        double d = 100.04;

        //explicit type casting
        long l = (long)d;

        //explicit type casting
        int i = (int)l;
        System.out.println("Double value "+d);

        //fractional part lost
        System.out.println("Long value "+l);

        //fractional part lost
        System.out.println("Int value "+i);
    }
}
```

Output:



```
Double value 100.04
Long value 100
Int value 100
```

3 Input & Output

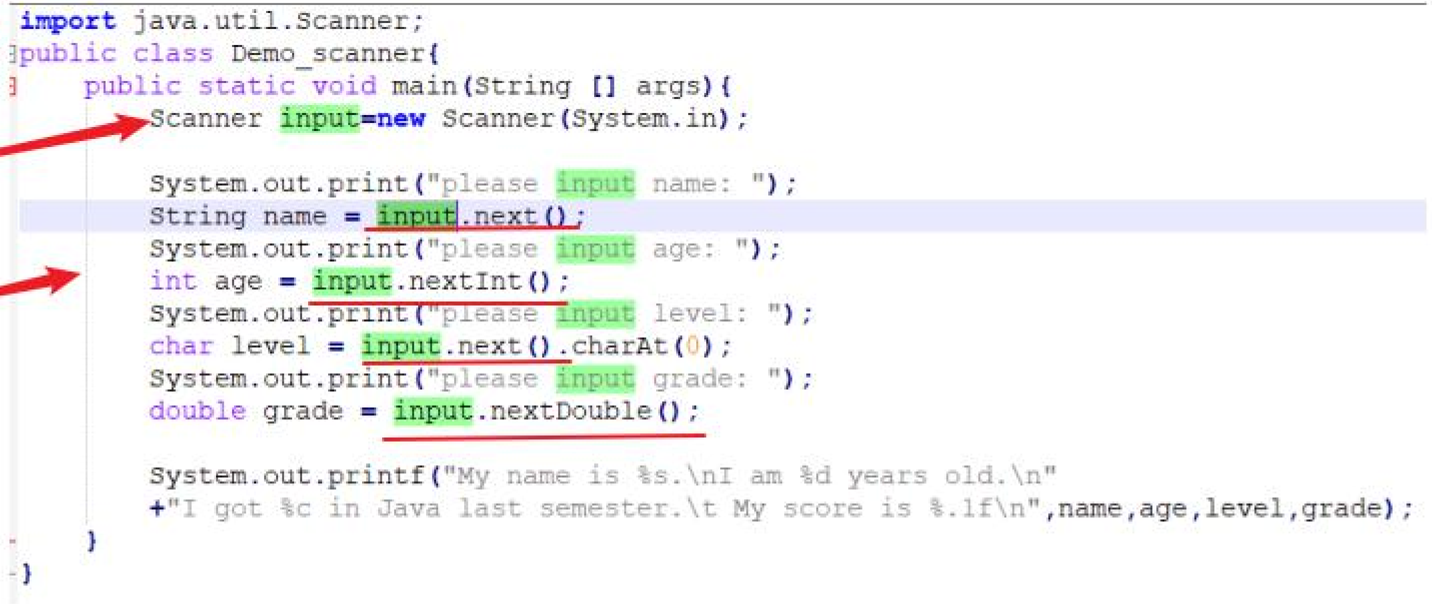


贾艳红 Jana
Email:jiayh@mail.sustech.edu.cn

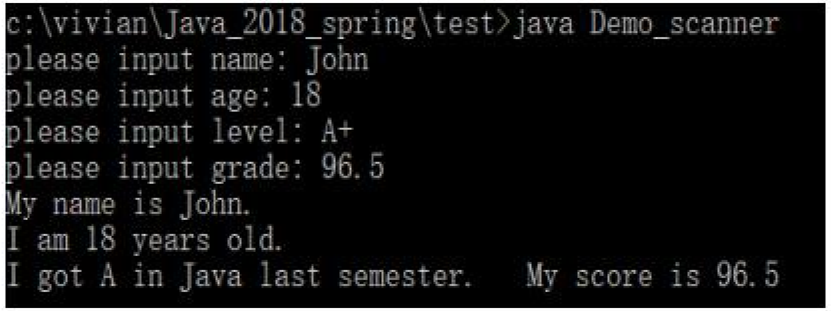
Using Scanner to Input

In you code

- 1. **import java.util.Scanner**
- 2. **new an Scanner object**
- 3. invoke the **method** of the Scanner object to get input data
  - 3-1: next() to get a string data
  - 3-2: nextInt() to get an integer data
  - 3-3: nextDouble() to get a double data
- 4. if all the input process is end, invoke the “**close()**” method is suggested to make your program safe



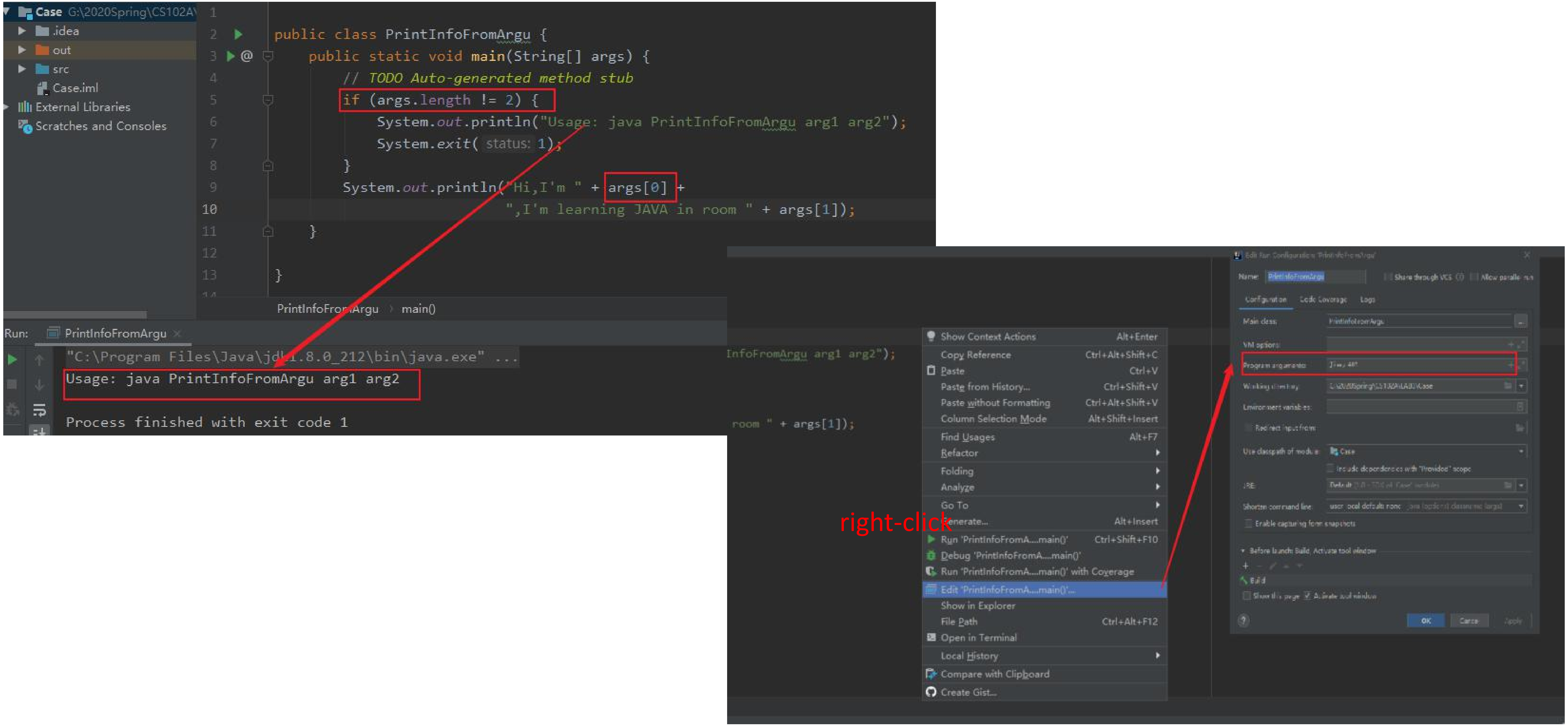
```java
import java.util.Scanner;
public class Demo_scanner{
    public static void main(String [] args){
        Scanner input=new Scanner(System.in);

        System.out.print("please input name: ");
        String name = input.next();
        System.out.print("please input age: ");
        int age = input.nextInt();
        System.out.print("please input level: ");
        char level = input.next().charAt(0);
        System.out.print("please input grade: ");
        double grade = input.nextDouble();

        System.out.printf("My name is %s.\nI am %d years old.\n"
        +"I got %c in Java last semester.\t My score is %.1f\n",name,age,level,grade);
    }
}
```



```
c:\vivian\Java_2018_spring\test>java Demo_scanner
please input name: John
please input age: 18
please input level: A+
please input grade: 96.5
My name is John.
I am 18 years old.
I got A in Java last semester.    My score is 96.5
```

Runing programs with arguments

- **In IDEA**



```
public class PrintInfoFromArgu {
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        if (args.length != 2) {
            System.out.println("Usage: java PrintInfoFromArgu arg1 arg2");
            System.exit( status: 1);
        }
        System.out.println("Hi,I'm " + args[0] +
                ",I'm learning JAVA in room " + args[1]);
    }

}
```

```
"C:\Program Files\Java\jdk1.8.0_212\bin\java.exe" ...
Usage: java PrintInfoFromArgu arg1 arg2

Process finished with exit code 1
```

Edit Run Configuration: 'PrintInfoFromArgu'

Name: PrintInfoFromArgu

Main class: PrintInfoFromArgu

Program arguments: Jiang 40

Working directory: G:\2020Spring\CS102A\LAB\Case

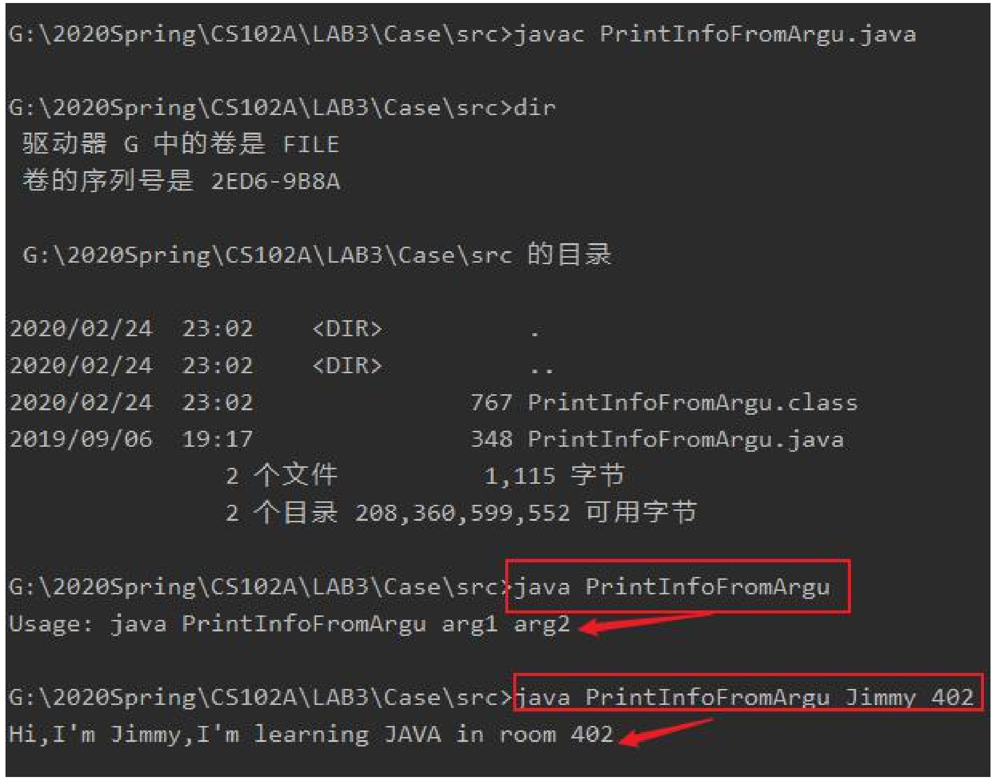
Use classpath of module: Case

right-click

Case G:\2020Spring\CS102A\
.idea
out
src
Case.iml
External Libraries
Scratches and Consoles
PrintInfoFromArgu › main()
Run: PrintInfoFromArgu
Show Context Actions Alt+Enter
Copy Reference Ctrl+Alt+Shift+C
Paste Ctrl+V
Paste from History... Ctrl+Shift+V
Paste without Formatting Ctrl+Alt+Shift+V
Column Selection Mode Alt+Shift+Insert
Find Usages Alt+F7
Refactor
Folding
Analyze
Go To
Generate... Alt+Insert
Run 'PrintInfoFromA....main()' Ctrl+Shift+F10
Debug 'PrintInfoFromA....main()'
Run 'PrintInfoFromA....main()' with Coverage
Edit 'PrintInfoFromA....main()'...
Show in Explorer
File Path Ctrl+Alt+F12
Open in Terminal
Local History
Compare with Clipboard
Create Gist...
Share through VCS
Allow parallel run
Configuration
Code Coverage
Logs
VM options:
Environment variables:
Redirect input from:
Include dependencies with "Provided" scope
JRE:
Shorten command line:
Enable capturing form snapshots
Before launch: Build, Activate tool window
Build
Show this page
Activate tool window
OK
Cancel
Apply

Runing programs with arguments

- **In command line**



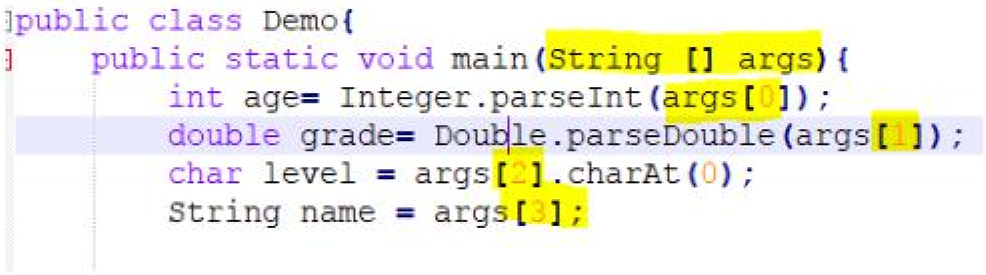
```
G:\2020Spring\CS102A\LAB3\Case\src>javac PrintInfoFromArgu.java

G:\2020Spring\CS102A\LAB3\Case\src>dir
 驱动器 G 中的卷是 FILE
 卷的序列号是 2ED6-9B8A

 G:\2020Spring\CS102A\LAB3\Case\src 的目录

2020/02/24  23:02    <DIR>          .
2020/02/24  23:02    <DIR>          ..
2020/02/24  23:02               767 PrintInfoFromArgu.class
2019/09/06  19:17               348 PrintInfoFromArgu.java
               2 个文件          1,115 字节
               2 个目录 208,360,599,552 可用字节

G:\2020Spring\CS102A\LAB3\Case\src>java PrintInfoFromArgu
Usage: java PrintInfoFromArgu arg1 arg2

G:\2020Spring\CS102A\LAB3\Case\src>java PrintInfoFromArgu Jimmy 402
Hi,I'm Jimmy,I'm learning JAVA in room 402
```

Conclusion

While using argument of running

1) The arguments is stored in String [] args

2) You may need to change its type as your desired



```
public class Demo{
    public static void main(String [] args){
        int age= Integer.parseInt(args[0]);
        double grade= Double.parseDouble(args[1]);
        char level = args[2].charAt(0);
        String name = args[3];
```

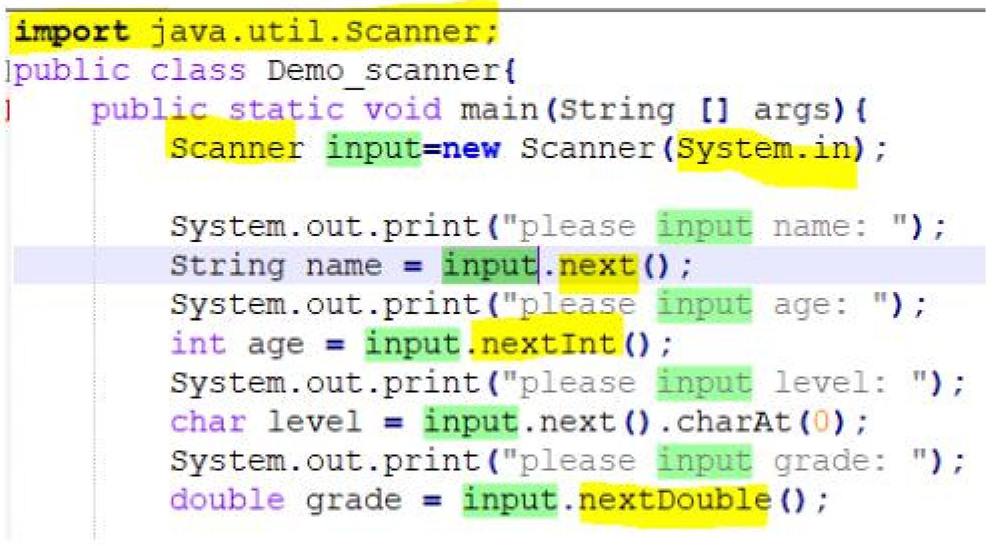
While using Scanner object

1) Import scanner

2) New a object

3) Invoke the method to get the right type of data

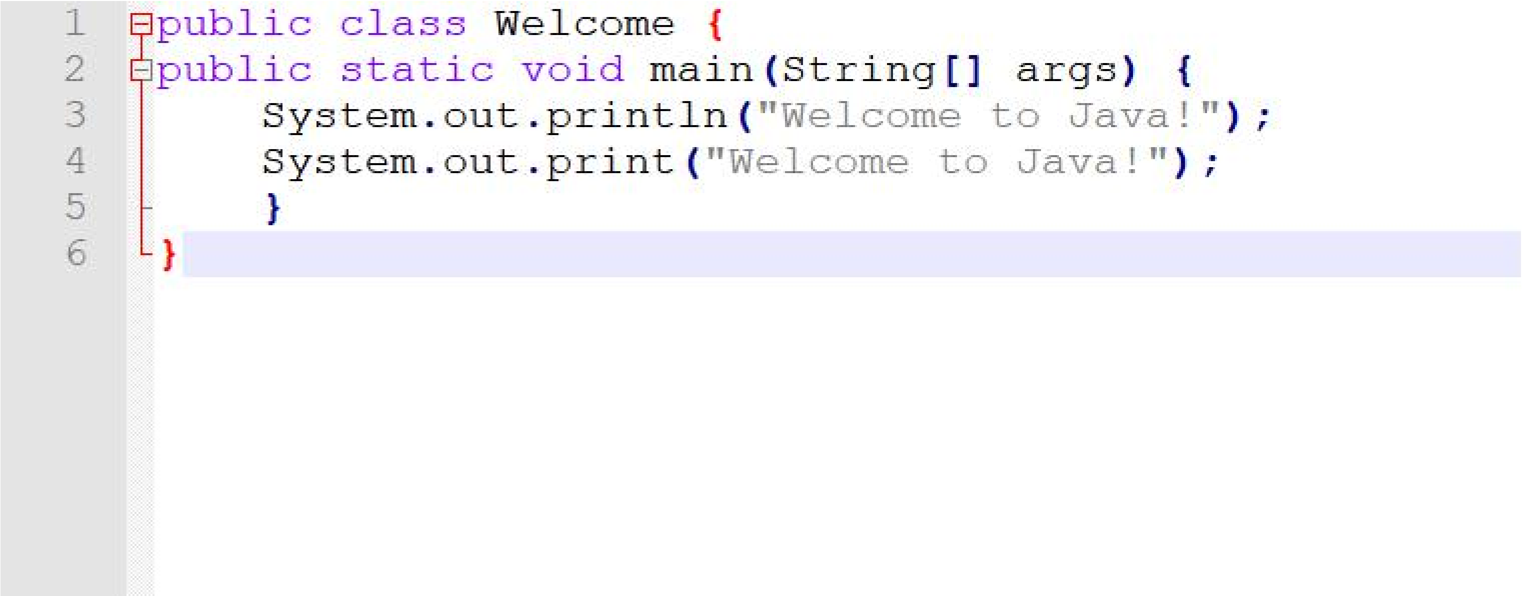
4) Invoke close() is sugguested



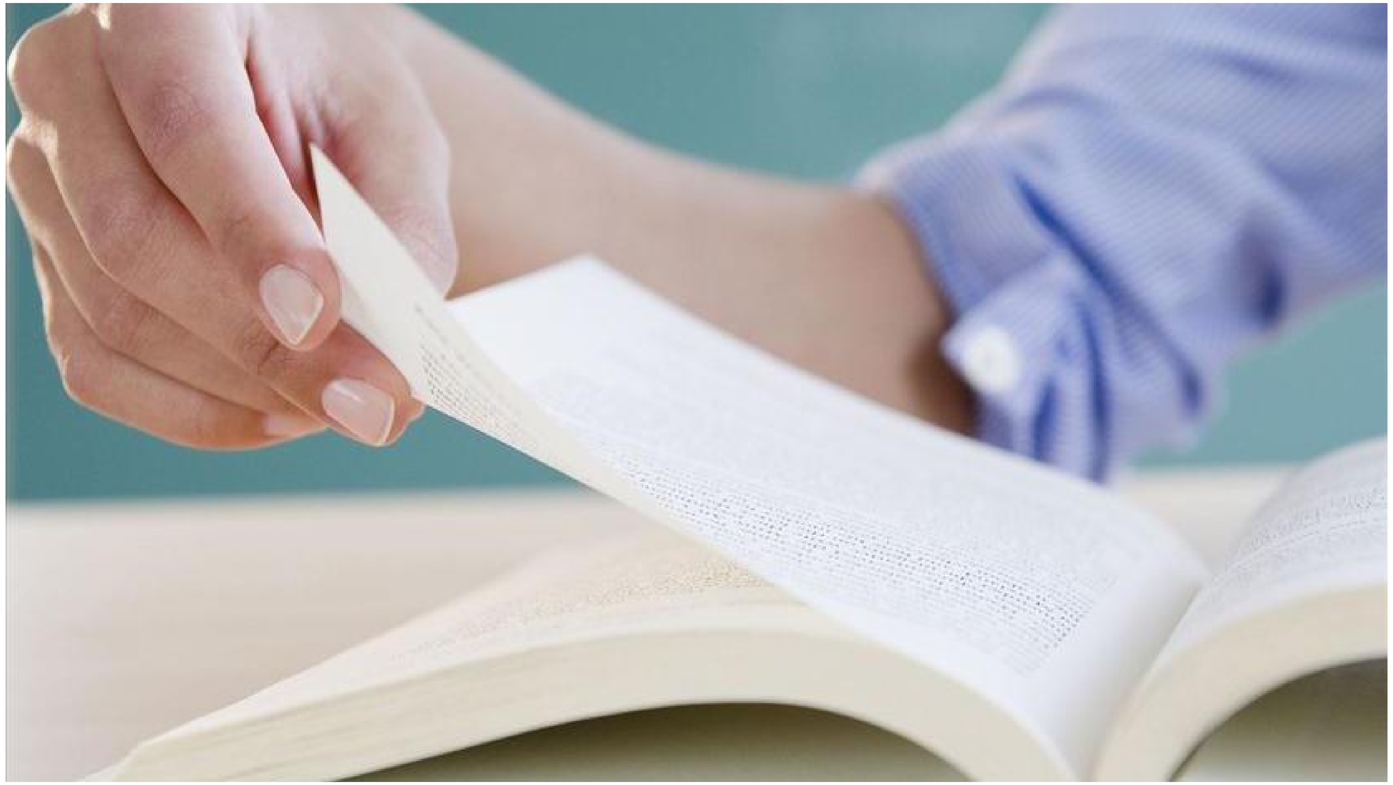
```
import java.util.Scanner;
public class Demo_scanner{
    public static void main(String [] args){
        Scanner input=new Scanner(System.in);

        System.out.print("please input name: ");
        String name = input.next();
        System.out.print("please input age: ");
        int age = input.nextInt();
        System.out.print("please input level: ");
        char level = input.next().charAt(0);
        System.out.print("please input grade: ");
        double grade = input.nextDouble();
```

Print without format

- System.out.println();
- System.out.print();



```java
public class Welcome {
public static void main(String[] args) {
     System.out.println("Welcome to Java!");
     System.out.print("Welcome to Java!");
     }
}
```



3 Selections

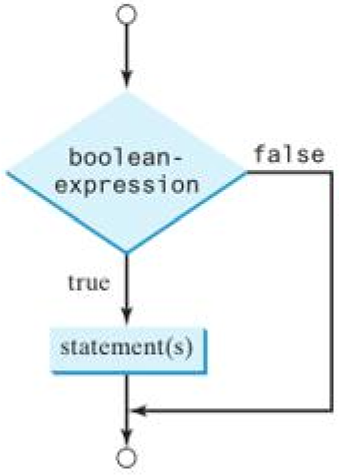
3.1 if Statements

3.2 if-else Statements

3.3 else-if Statements

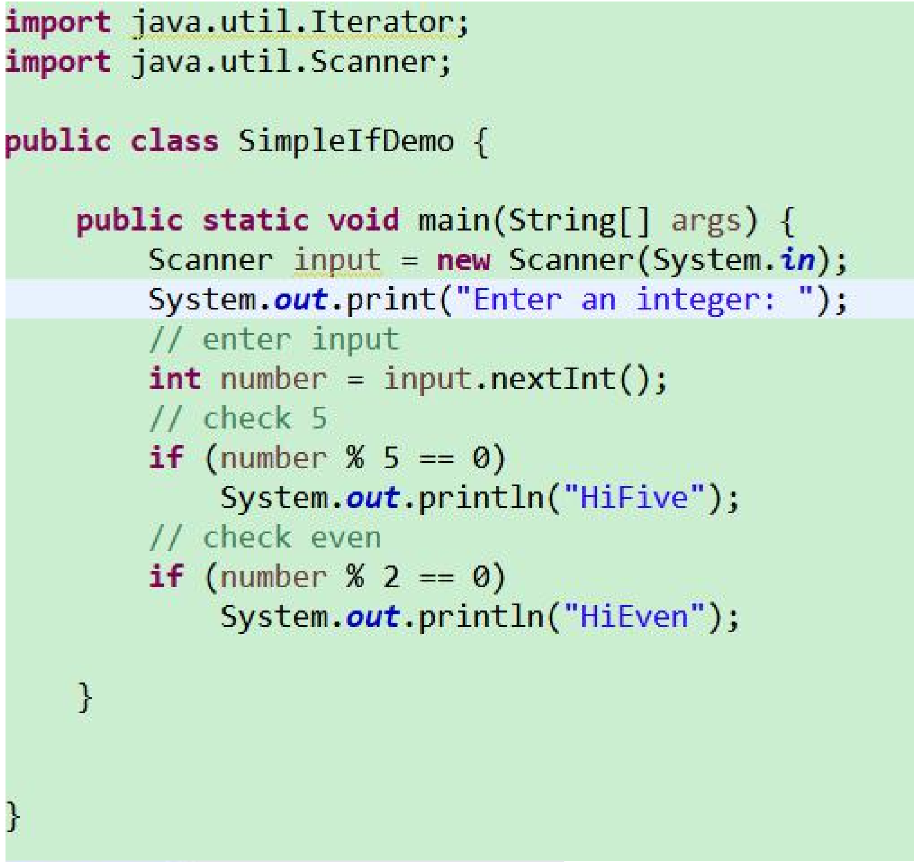
贾艳红 Jana
Email:jiayh@mail.sustech.edu.cn

3.1 if Statements(if)

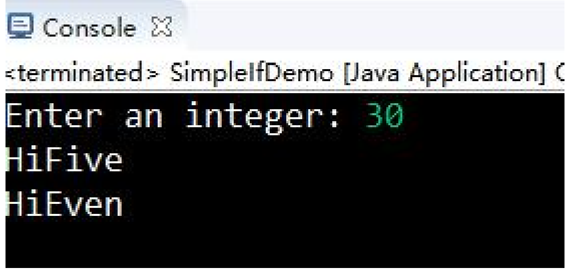


```
if (boolean-expression) {
  statement(s);
}
```

Example:

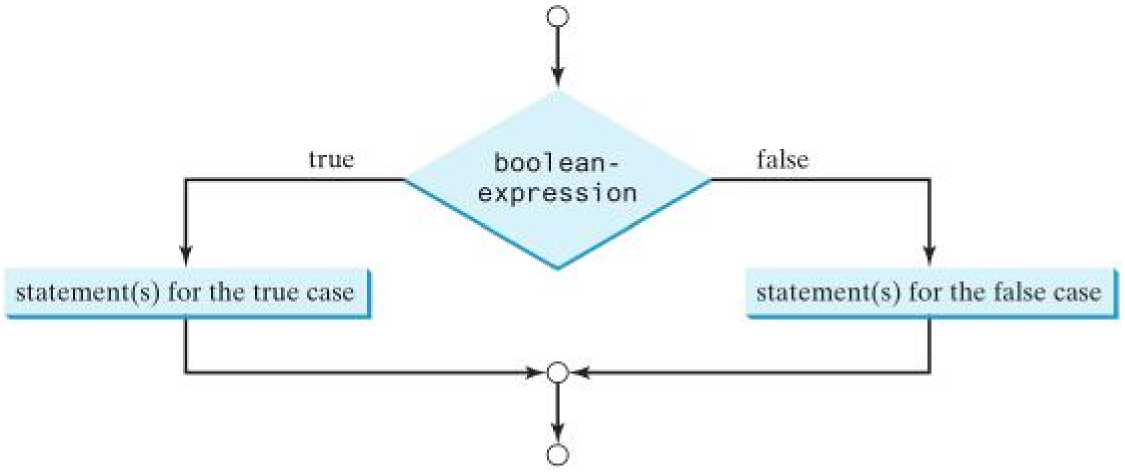


```
import java.util.Iterator;
import java.util.Scanner;

public class SimpleIfDemo {

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print("Enter an integer: ");
        // enter input
        int number = input.nextInt();
        // check 5
        if (number % 5 == 0)
            System.out.println("HiFive");
        // check even
        if (number % 2 == 0)
            System.out.println("HiEven");

    }


}
```



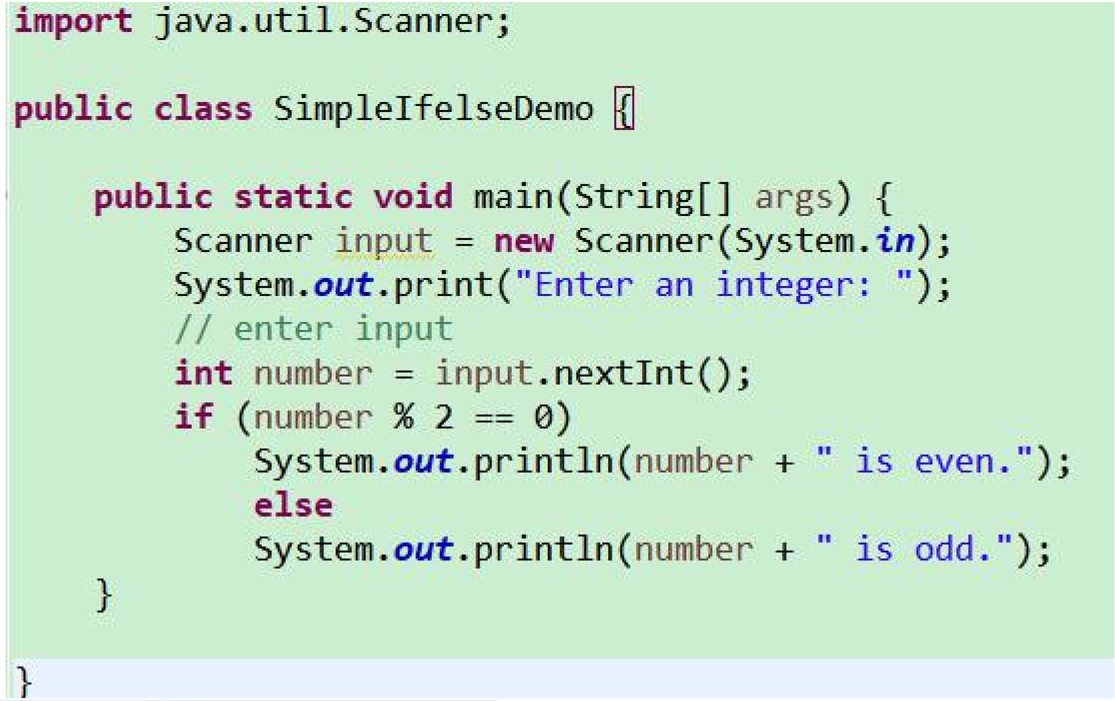
```
<terminated> SimpleIfDemo [Java Application] C
Enter an integer: 30
HiFive
HiEven
```

3.2 if Statements(if-else)

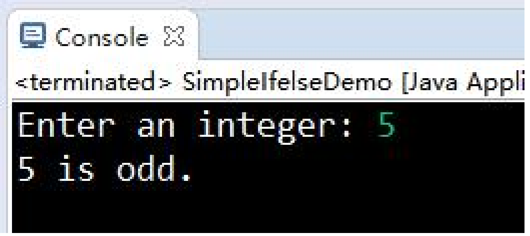

true
boolean-expression
false
statement(s) for the true case
statement(s) for the false case

```
if (boolean-expression) {
  statement(s)-for-the-true-case;
}
else {
  statement(s)-for-the-false-case;
}
```

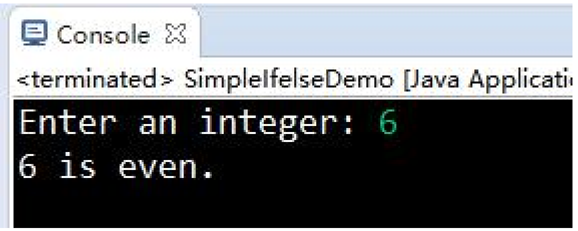
Example:



```
import java.util.Scanner;

public class SimpleIfelseDemo {

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print("Enter an integer: ");
        // enter input
        int number = input.nextInt();
        if (number % 2 == 0)
            System.out.println(number + " is even.");
            else
            System.out.println(number + " is odd.");
    }

}
```

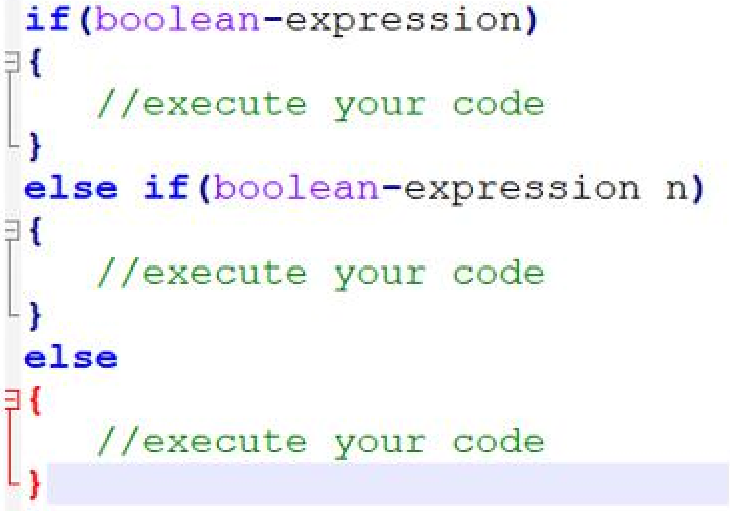


```
<terminated> SimpleIfelseDemo [Java Appli
Enter an integer: 5
5 is odd.
```



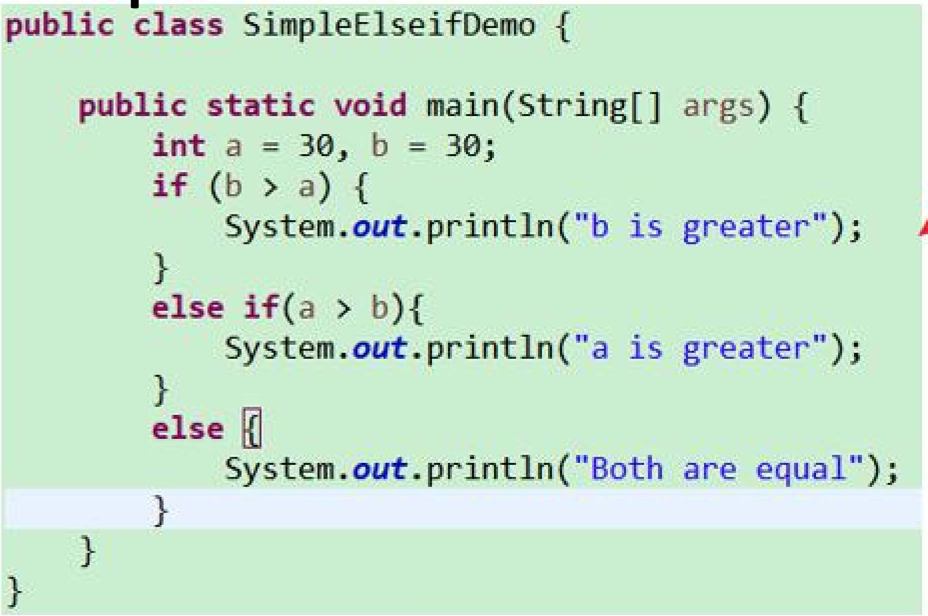
```
<terminated> SimpleIfelseDemo [Java Applicati
Enter an integer: 6
6 is even.
```

3.3 if Statements(else-if and Nested if)



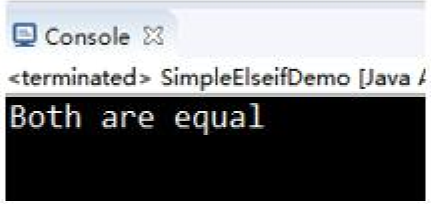
```
if(boolean-expression)
{
    //execute your code
}
else if(boolean-expression n)
{
    //execute your code
}
else
{
    //execute your code
}
```

Example:

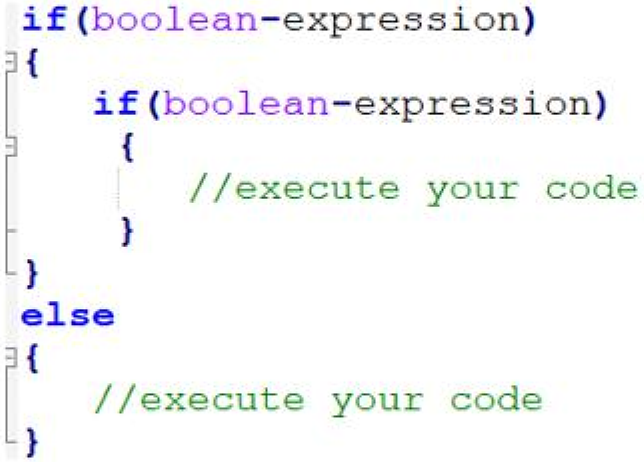


```
public class SimpleElseifDemo {

    public static void main(String[] args) {
        int a = 30, b = 30;
        if (b > a) {
            System.out.println("b is greater");
        }
        else if(a > b){
            System.out.println("a is greater");
        }
        else {
            System.out.println("Both are equal");
        }
    }
}
```

else-if

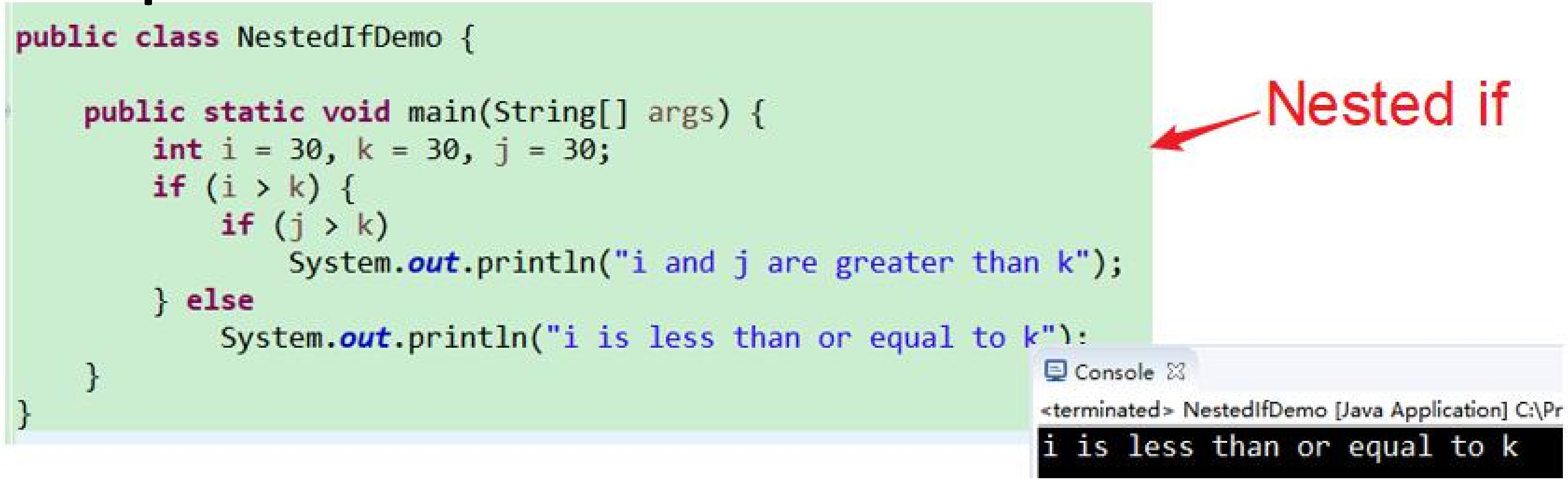


```
<terminated> SimpleElseifDemo [Java A
Both are equal
```



```
if(boolean-expression)
{
    if(boolean-expression)
     {
        //execute your code
     }
}
else
{
    //execute your code
}
```

Example:

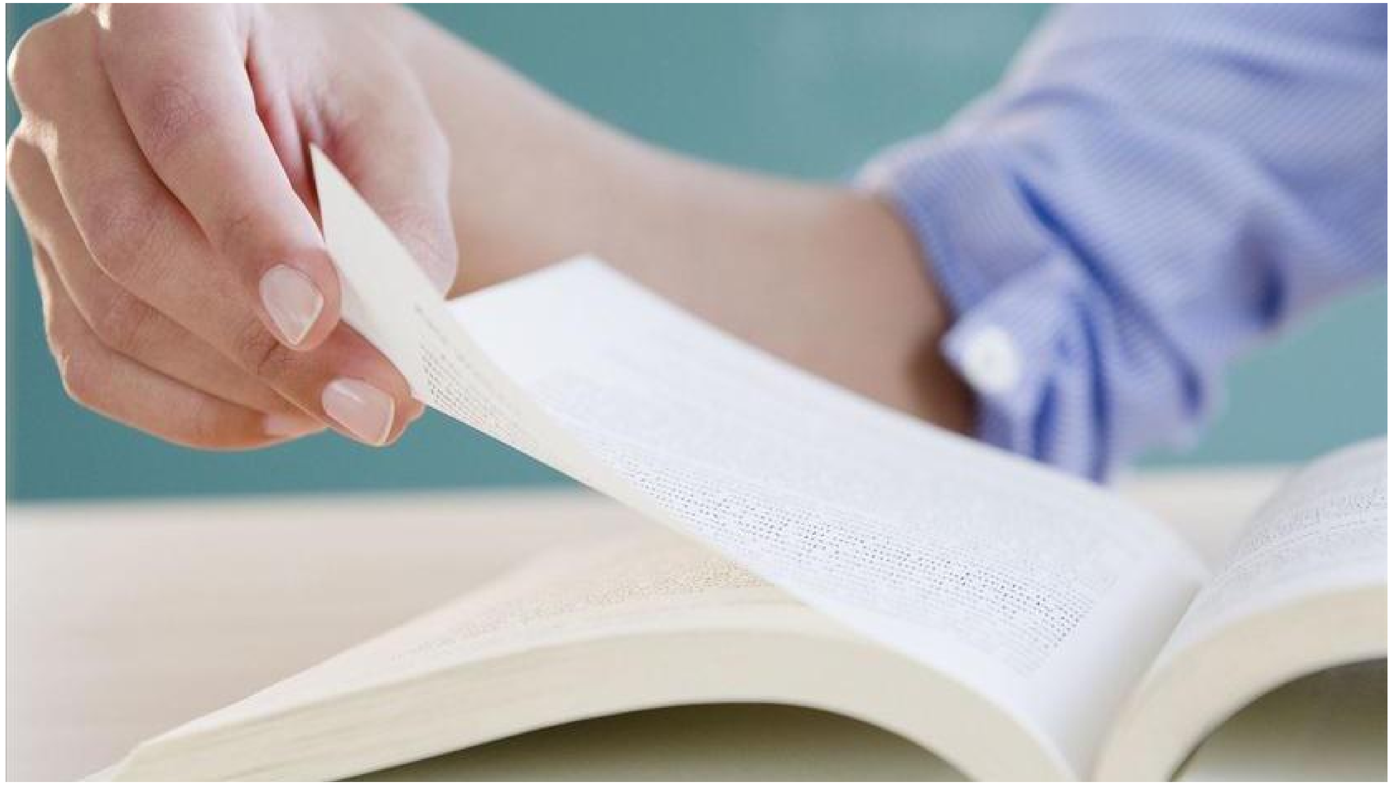


```
public class NestedIfDemo {

    public static void main(String[] args) {
        int i = 30, k = 30, j = 30;
        if (i > k) {
            if (j > k)
                System.out.println("i and j are greater than k");
        } else
            System.out.println("i is less than or equal to k");
    }
}
```

```
<terminated> NestedIfDemo [Java Application] C:\Pr
i is less than or equal to k
```

Nested if

3 Exercises



Complete the exercises in the **2021S-Java-A-Lab-2.pdf and 2021S-Java-A-Lab-2_Part2_.pdf** .

THANK YOU

贾艳红 Jana
Email:jiayh@mail.sustech.edu.cn