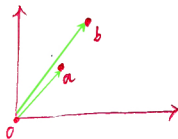


Object-Oriented Programming in Java



$$\vec{oa} + \vec{ob} = \vec{oc}$$

$$a = (x_a, y_a)$$

$$b = (x_b, y_b)$$

$$c = a + b = (x_a + x_b, y_a + y_b)$$

$$= (x_c, y_c)$$

1. Open a new file named "Point.java";

```
Point.java
1 package Algebra;
2
3 public class Point
4 {
5     public float x;
6     public float y;
7     public Point(float vx, float vy)
8     {
9         this.x = vx;
10        this.y = vy;
11    }
12 }
```

2. Open a new file named "Add.java";

```
Add.java
1 package Algebra;
2 import Algebra.Point;
3
4 public class Add
5 {
6     public static void main(String args[])
7     {
8         Point a = new Point( 1.1f, 2.2f);
9         Point b = new Point( 1.7f, 3.0f);
10        Point c = new Point( a.x+b.x , a.y+b.y );
11        System.out.println( "Point_a + Point_b = (" + c.x + "," + c.y + ")" );
12    }
13 }
```

3. The whole process should be looked like:

```
Windows PowerShell
Windows PowerShell
版权所有 (C) Microsoft Corporation. 保留所有权利。
尝试新的跨平台 PowerShell https://aka.ms/powershell

PS C:\Users\luomi> cd D:/
PS D:\> mkdir Code

目录: D:\

Mode                LastWriteTime         Length Name
----                -
d-----          2023/10/14   16:22             Code

PS D:\> cd Code
PS D:\Code> mkdir Algebra

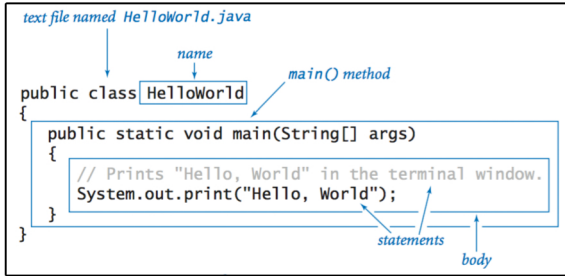
目录: D:\Code

Mode                LastWriteTime         Length Name
----                -
d-----          2023/10/14   16:22             Algebra

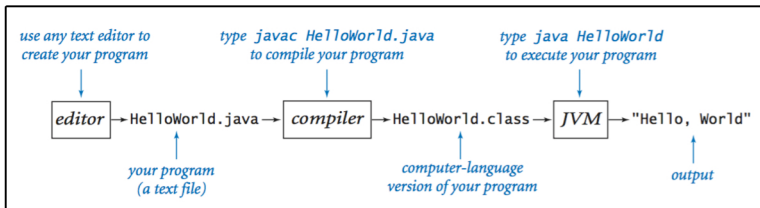
PS D:\Code> javac Algebra/Add.java Algebra/Point.java
PS D:\Code> java Algebra.Add

Point_a + Point_b (2.8000002 5.2)
```

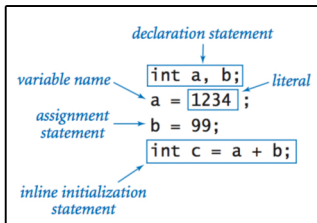
Hello, World.



Editing, compiling, and executing.



Declaration and assignment statements.



Java Built-in data types.

type	set of values	common operators	sample literal values
int	integers	+ - * / %	99 12 2147483647
double	floating-point numbers	+ - * /	3.14 2.5 6.022e23
boolean	boolean values	&& !	true false
char	characters		'A' '1' '%' '\n'
String	sequences of characters	+	"AB" "Hello" "2.5"