

Panel 1

Program: ~~g~~ to instruct a computer to solve a task

Programming Language: foreign language used to instruct computer. Typed as "plain text", is readable by humans, vocabulary + grammar.

Compiling: check grammar + vocab and translate it to "machine language"

Executing: allocate resources + perform steps outlined in the program

The mechanics:  
 IDE      Edit      Compile      execute  
           Source

1

Panel 2

Task:  $1 + 2 + 3 + \dots + 99 + 100 = \frac{100 \times 101}{2}$

$100 + 99 + 98 + \dots + 2 + 1 = \nearrow$

$101 + 101 + 101 + \dots + 101 + 101 = 100 \times 101$

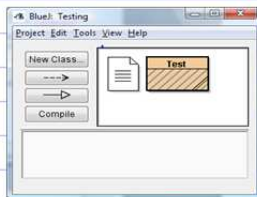
2

Panel 3

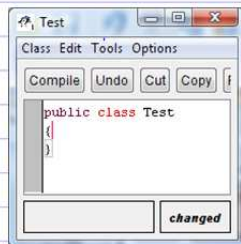
Download + install JDK and BlueJ. Then start BlueJ



① Click on "Project" to create a new project. Use a name without spaces as project name



② Create a new class named "Test" and double-click to open the editor



③ Erase everything except  
 public class Test  
 {  
 }  
 Then compile - no errors

3

Panel 4

#### Definition 1.01: Basic Java Programming Guidelines

Every Java program must follow these guidelines:

- Java is case sensitive, i.e. the word `Program` is different from `program`.
- Curly brackets `{` and `}` are used to group statements together.
- An executable Java program must contain at least the following lines as a framework:

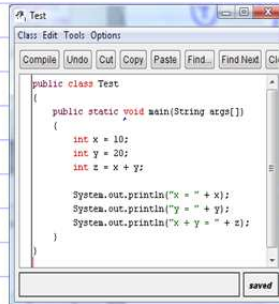
```
public class Name
{
    public static void main(String args[])
    {
        ... program code ...
    }
}
```

- Every statement whose next statement is not a separate group must end in a semicolon.
- A Java program containing the above framework must be saved using the filename `Name.java`, where `Name` (including correct upper and lower cases) is the word that follows the keywords `public class` and the file extension is `.java`.

4

Panel 5

④ Type the following and compile :

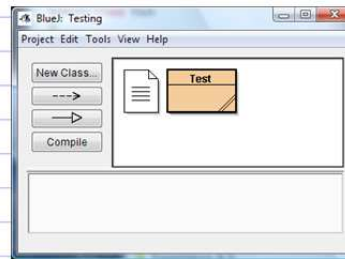


```
public class Test
{
    public static void main(String args[])
    {
        int x = 10;
        int y = 20;
        int z = x + y;

        System.out.println("x = " + x);
        System.out.println("y = " + y);
        System.out.println("x + y = " + z);
    }
}
```

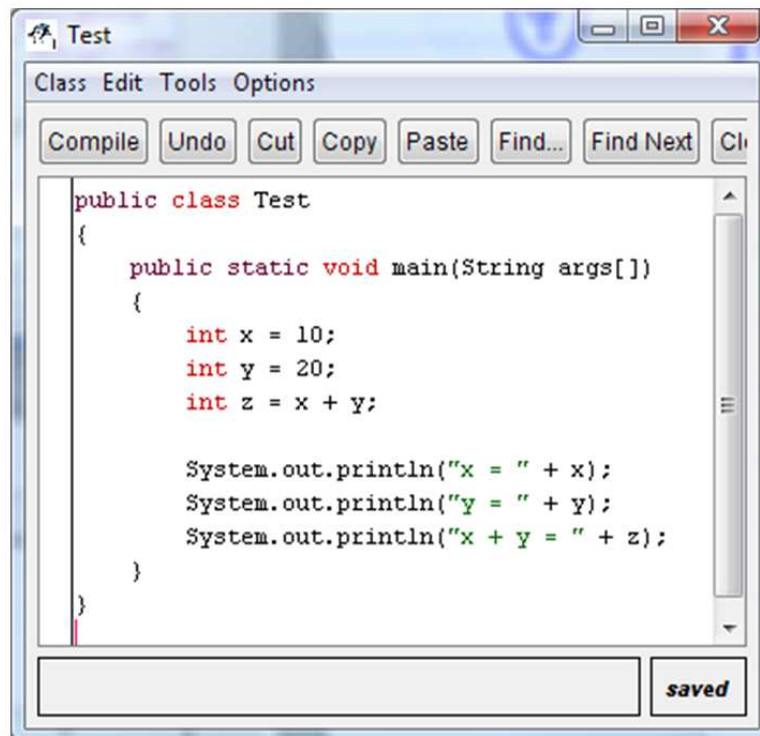
⑤ Execute by going back to Project window.

Right-click on the  
"Test" class and select  
the "main" method



5

Panel 6



```
public class Test
{
    public static void main(String args[])
    {
        int x = 10;
        int y = 20;
        int z = x + y;

        System.out.println("x = " + x);
        System.out.println("y = " + y);
        System.out.println("x + y = " + z);
    }
}
```

6

Panel 7

Create new class named: Rings  
and type the following:

```
import java.awt.*;
import java.awt.event.*;

public class Rings extends Frame
{
    public Rings()
    {
        setTitle("Rings");
        setSize(300, 200);
        setVisible(true);
    }

    public static void main(String args[])
    {
        System.out.println("Starting Rings program");
        Rings rings = new Rings();
    }
}
```

7

Panel 8

```
import java.awt.*;
import java.awt.event.*;

public class Rings extends Frame
{
    private class WindowCloser extends WindowAdapter
    {
        public void windowClosing(WindowEvent we)
        {
            System.exit(0);
        }
    }

    public Rings()
    {
        setTitle("Rings");
        setSize(300, 200);
        setVisible(true);
        addWindowListener(new WindowCloser());
    }

    public void paint(Graphics g)
    {
        g.setColor(Color.red);
        g.drawOval(10, 30, 30, 30);
        g.setColor(Color.blue);
        g.drawOval(35, 30, 30, 30);
        g.setColor(Color.green);
        g.drawOval(60, 30, 30, 30);
        g.setColor(Color.yellow);
        g.drawOval(85, 30, 30, 30);
        g.setColor(Color.black);
        g.drawOval(110, 30, 30, 30);
        g.drawString("Rings", 40, 100);
    }

    public static void main(String args[])
    {
        System.out.println("Starting Rings program");
        Rings rings = new Rings();
    }
}
```

8

Panel 9

Homework:

Either:



Olympics 08

Or:



Snowman