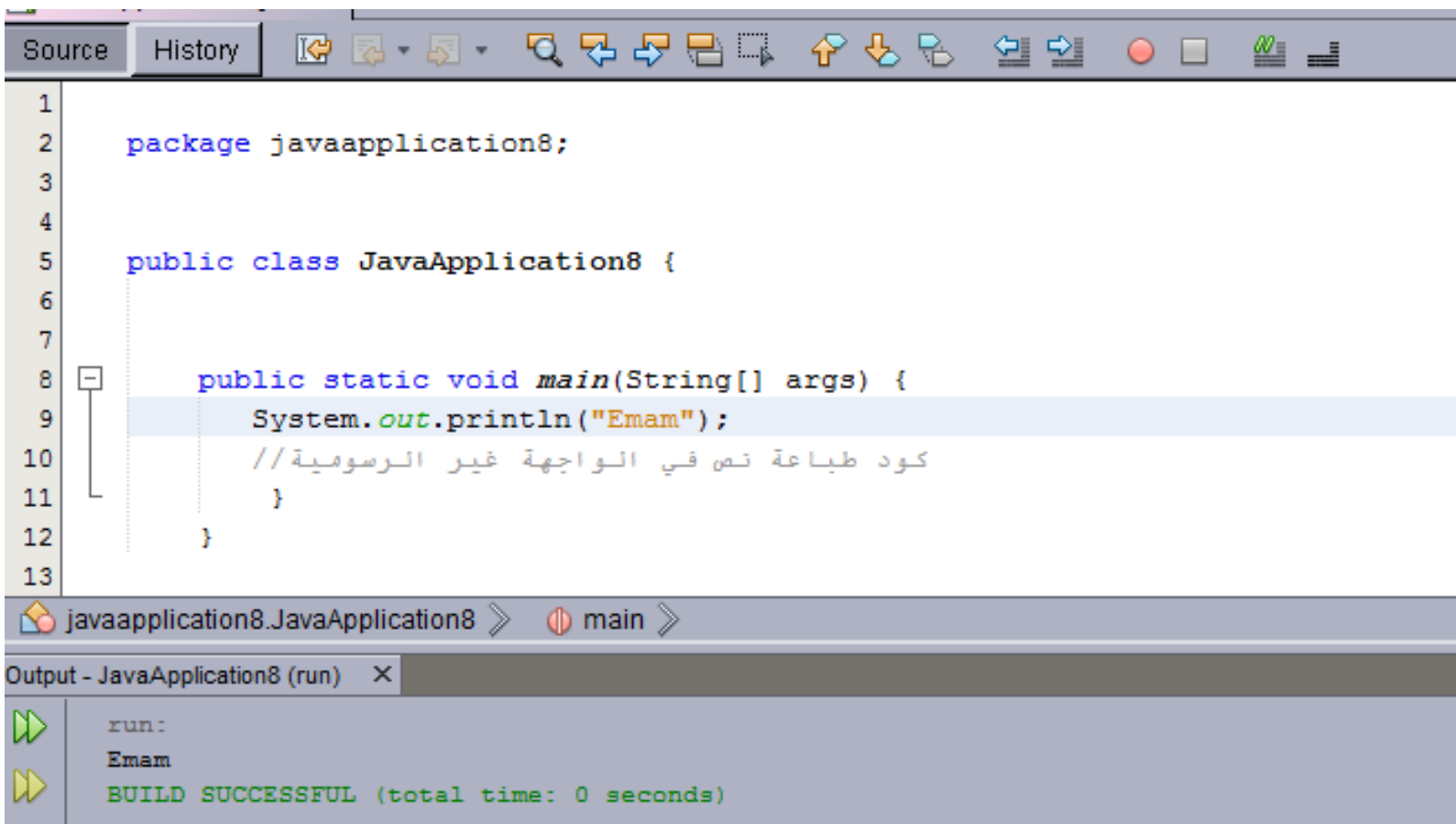


1- لطباعه نص :



The screenshot shows an IDE window with a source code editor and an output console. The source code editor displays a Java program that prints "Emam" to the console. The output console shows the result of running the program, which is "Emam", and a message indicating that the build was successful.

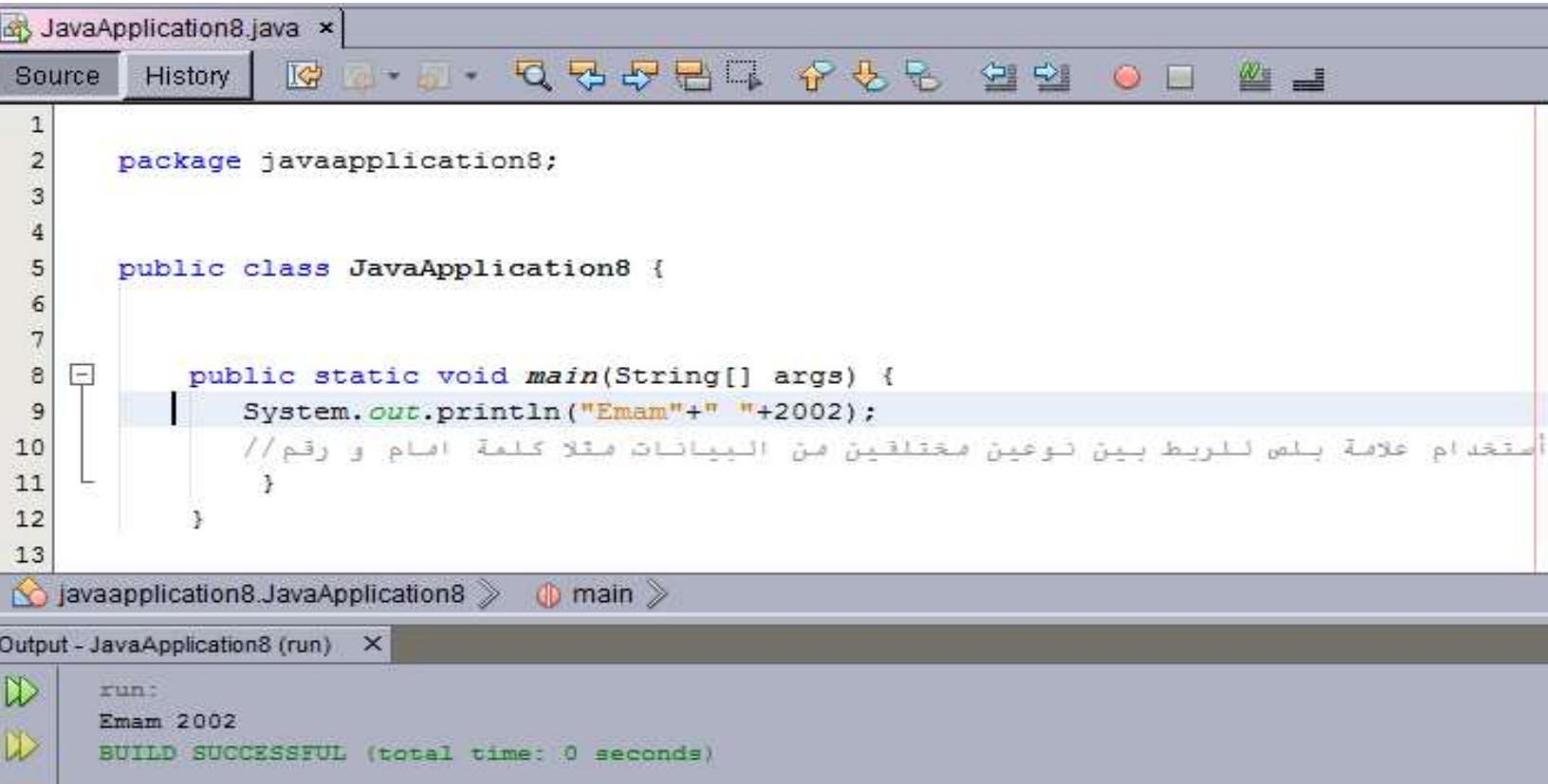
```
1
2 package javaapplication8;
3
4
5 public class JavaApplication8 {
6
7
8     public static void main(String[] args) {
9         System.out.println("Emam");
10        // كود طباعة نص في الواجهة غير الرسومية
11    }
12
13 }
```

javaapplication8.JavaApplication8 > main >

Output - JavaApplication8 (run) X

run:
Emam
BUILD SUCCESSFUL (total time: 0 seconds)

2- علامة +



The screenshot shows an IDE window titled "JavaApplication8.java". The code is as follows:

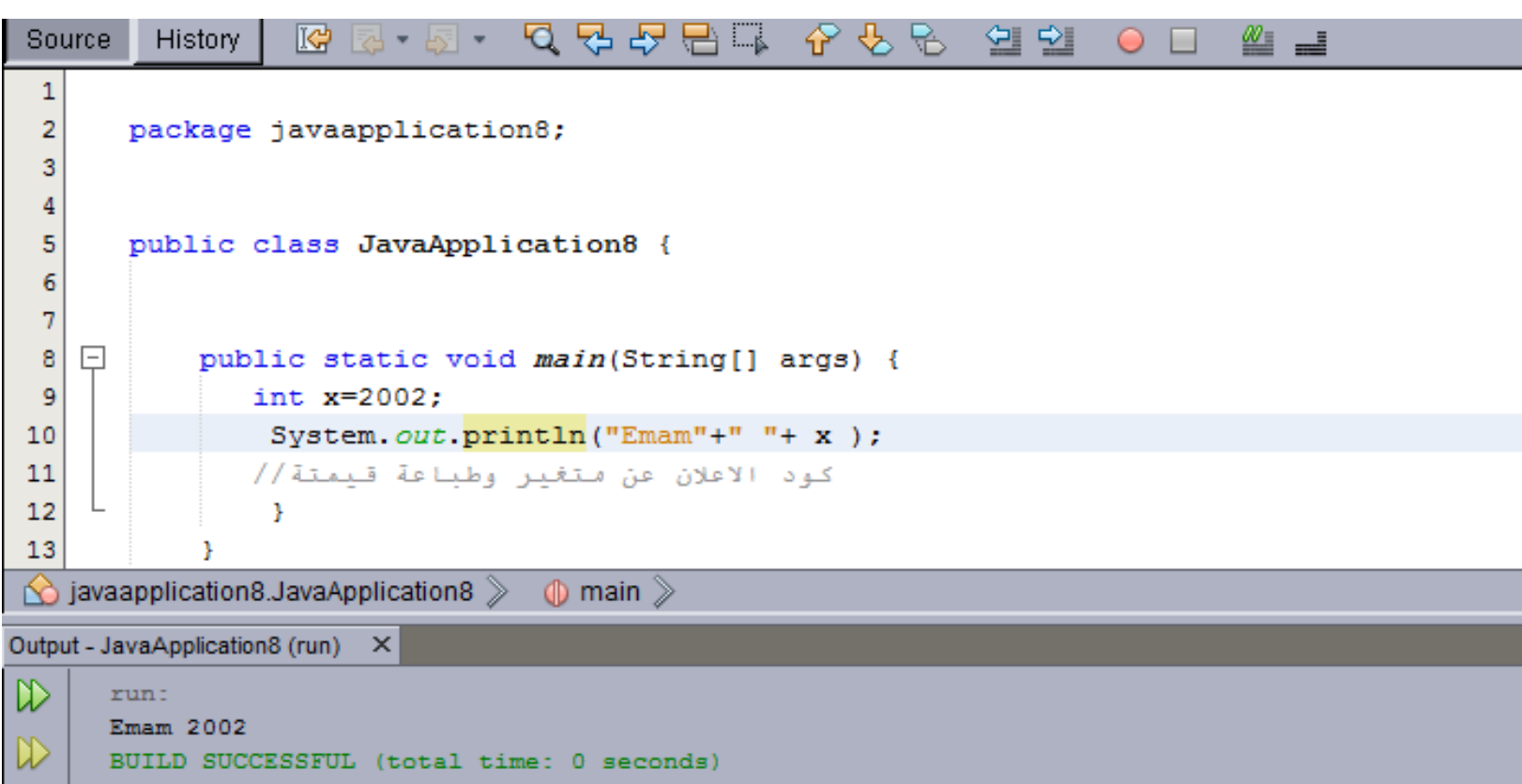
```
1 package javaapplication8;
2
3
4
5 public class JavaApplication8 {
6
7
8     public static void main(String[] args) {
9         System.out.println("Emam"+" "+2002);
10        // استخدام علامة بلمس للربط بين نوعين مختلفين من البيانات مثلا كلمة امام و رقم
11    }
12 }
13
```

The IDE's breadcrumb navigation shows the path: `javaapplication8.JavaApplication8` > `main`.

The "Output - JavaApplication8 (run)" window at the bottom shows the following output:

```
run:
Emam 2002
BUILD SUCCESSFUL (total time: 0 seconds)
```

3- طباعة قيمة متغير



The screenshot displays an IDE window with a source code editor and an output console. The source code editor shows a Java program with a package declaration, a class definition, and a main method. The main method contains a variable declaration, a print statement, and a comment in Arabic. The output console shows the execution of the program, displaying the output 'Emam 2002' and a successful build message.

```
1
2 package javaapplication8;
3
4
5 public class JavaApplication8 {
6
7
8     public static void main(String[] args) {
9         int x=2002;
10        System.out.println("Emam"+" "+ x );
11        // كود الاعلان عن متغير وطباعة قيمته
12    }
13 }
```

javaapplication8.JavaApplication8 > main >

Output - JavaApplication8 (run) X

run:
Emam 2002
BUILD SUCCESSFUL (total time: 0 seconds)

4- برنامج لجمع عددين



The screenshot shows an IDE with a source code editor and an output console. The source code is a Java application that adds two numbers, 10 and 20, and prints the result, 30. The output console shows the result of the run, which is 30, and a successful build message.

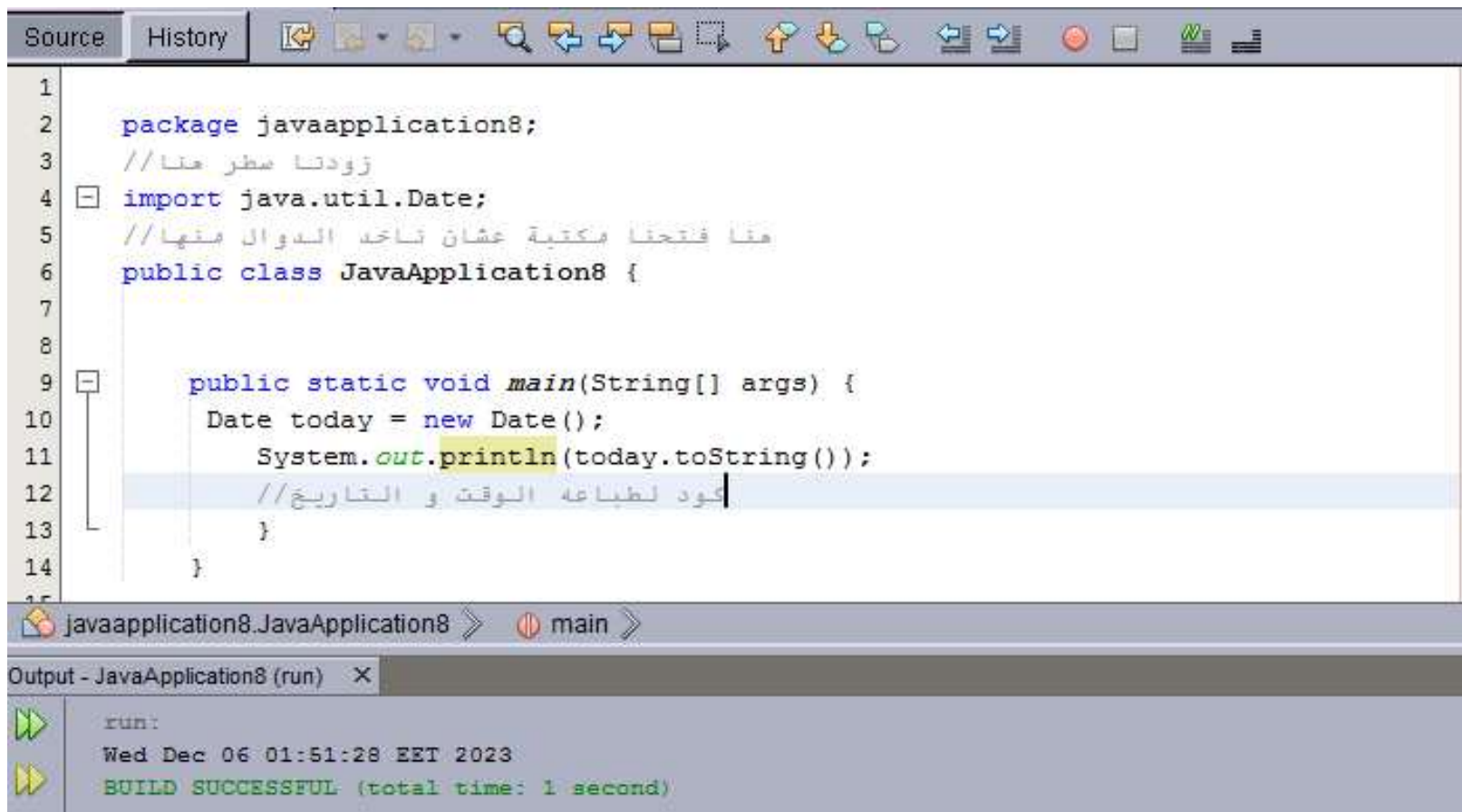
```
1
2 package javaapplication8;
3
4
5 public class JavaApplication8 {
6
7
8     public static void main(String[] args) {
9         int num1, num2, result;
10        num1=10; num2=20;
11        result=num1+num2;
12        System.out.println(result);
13        // كود جمع عددين
14        // ممكن قسمة أو ضرب أو طرح يس غير المعامل الخاص بالجمع
15    }
16 }
```

javaapplication8.JavaApplication8 > main >

Output - JavaApplication8 (run) X

run:
30
BUILD SUCCESSFUL (total time: 0 seconds)

5- كود طباعه الوقت والتاريخ



The screenshot shows an IDE window with a source code editor and an output console. The source code is a Java application named `JavaApplication8` in the package `javaapplication8`. It imports `java.util.Date` and contains a `main` method that creates a `Date` object and prints its string representation. The output console shows the result of running the application, displaying the current date and time: `Wed Dec 06 01:51:28 EET 2023`. The build was successful.

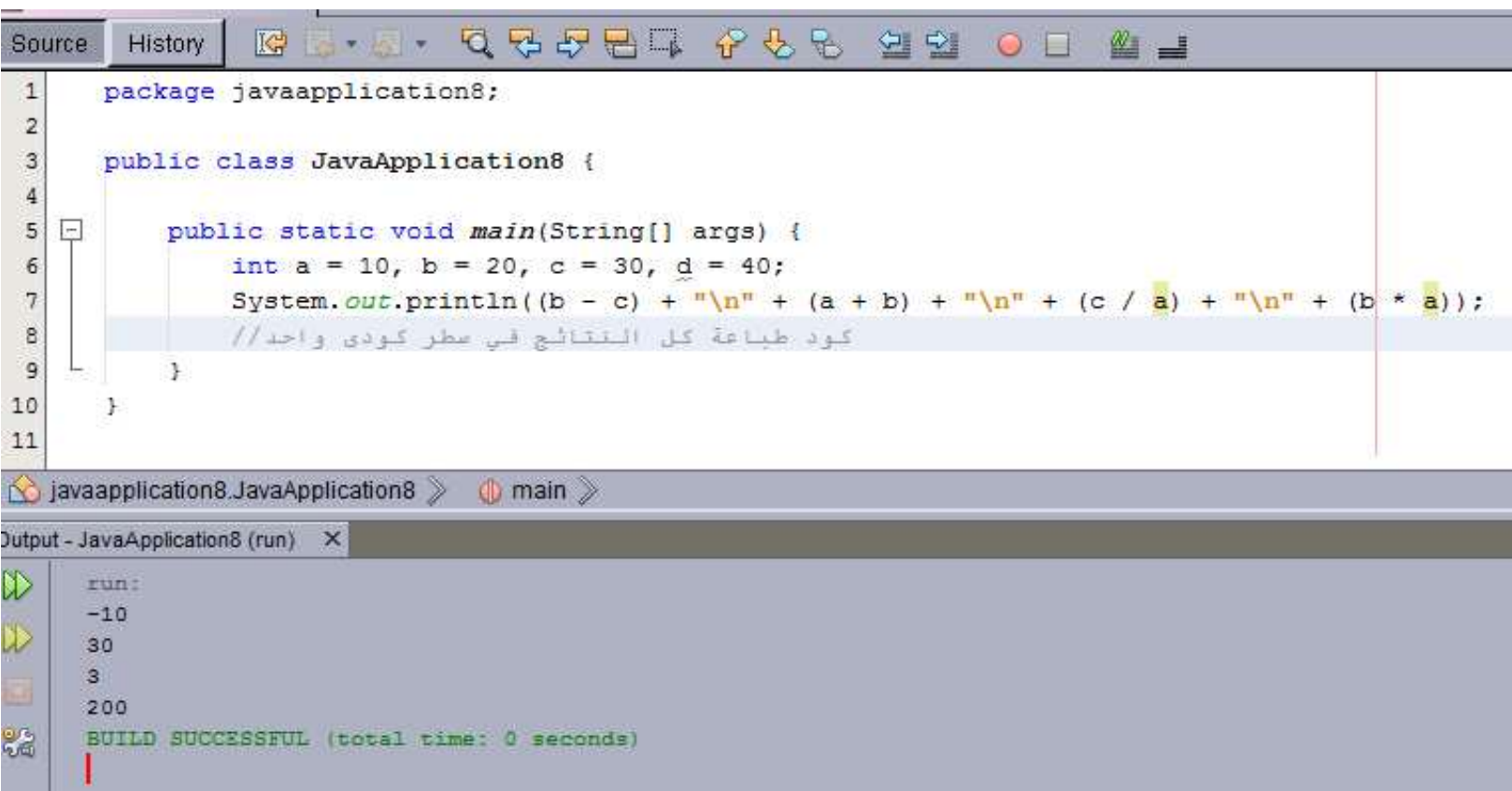
```
1 package javaapplication8;
2 //زودنا سطر هنا
3
4 import java.util.Date;
5 //هنا فتحنا مكتبة عشان تاخذ الدوال منها
6 public class JavaApplication8 {
7
8
9     public static void main(String[] args) {
10         Date today = new Date();
11         System.out.println(today.toString());
12         //كود لطباعه الوقت و التاريخ
13     }
14 }
```

javaapplication8.JavaApplication8 > main >

Output - JavaApplication8 (run) X

run:
Wed Dec 06 01:51:28 EET 2023
BUILD SUCCESSFUL (total time: 1 second)

6- كود طباعه أكثر من ناتج في سطر واحد



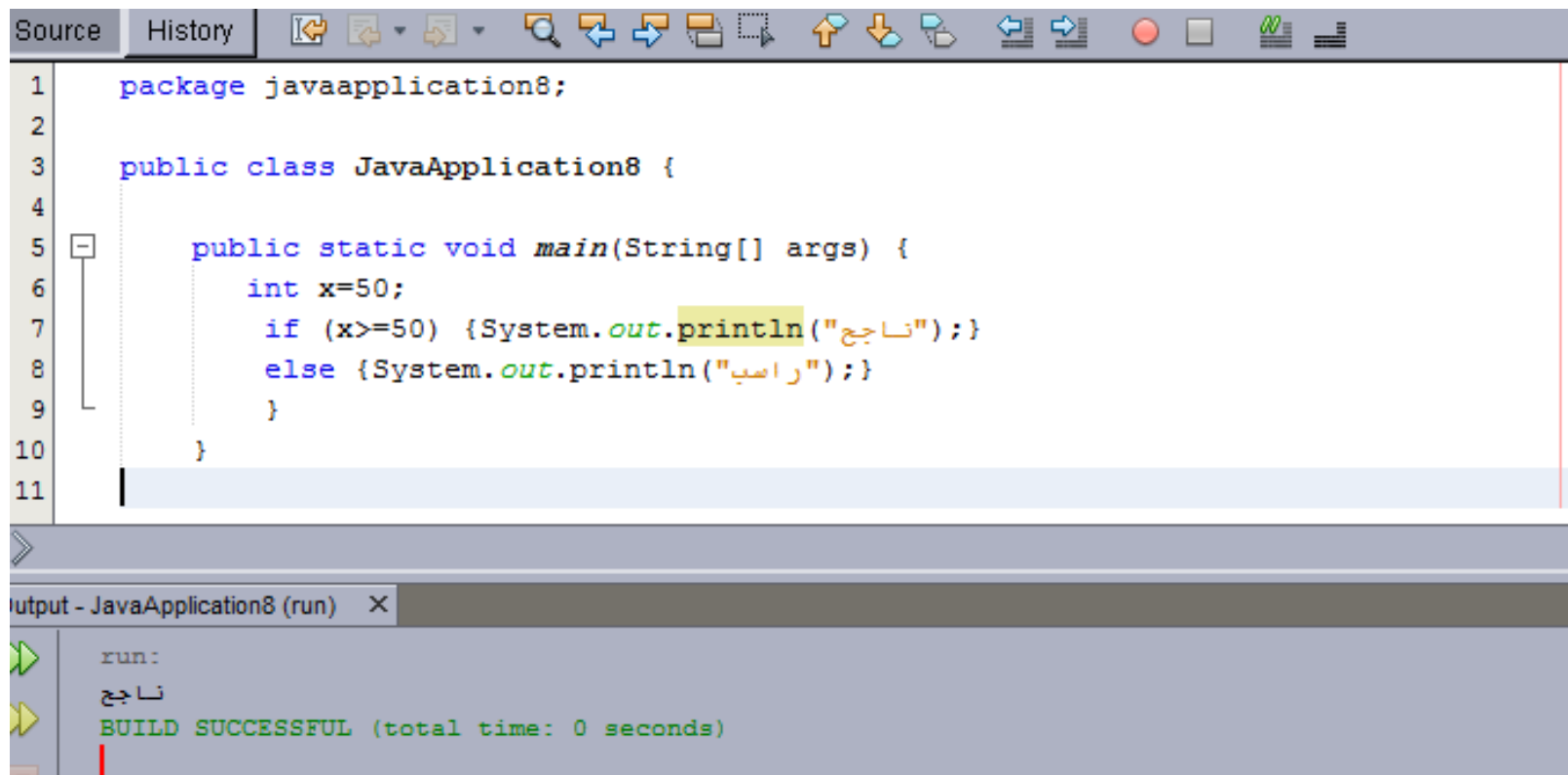
The screenshot shows an IDE with a Java source file and its output. The source file, `JavaApplication8.java`, contains a `main` method that calculates and prints the results of several arithmetic operations on variables `a`, `b`, `c`, and `d`. The output window shows the results of these operations, each on a new line, separated by `\n` characters in the code. The output is: `-10`, `30`, `3`, and `200`. The build is successful.

```
1 package javaapplication8;
2
3 public class JavaApplication8 {
4
5     public static void main(String[] args) {
6         int a = 10, b = 20, c = 30, d = 40;
7         System.out.println((b - c) + "\n" + (a + b) + "\n" + (c / a) + "\n" + (b * a));
8         // كود طباعة كل النتائج في سطر كودى واحد
9     }
10 }
11
```

Output - JavaApplication8 (run)

```
run:
-10
30
3
200
BUILD SUCCESSFUL (total time: 0 seconds)
```

7- جمله الشرط if



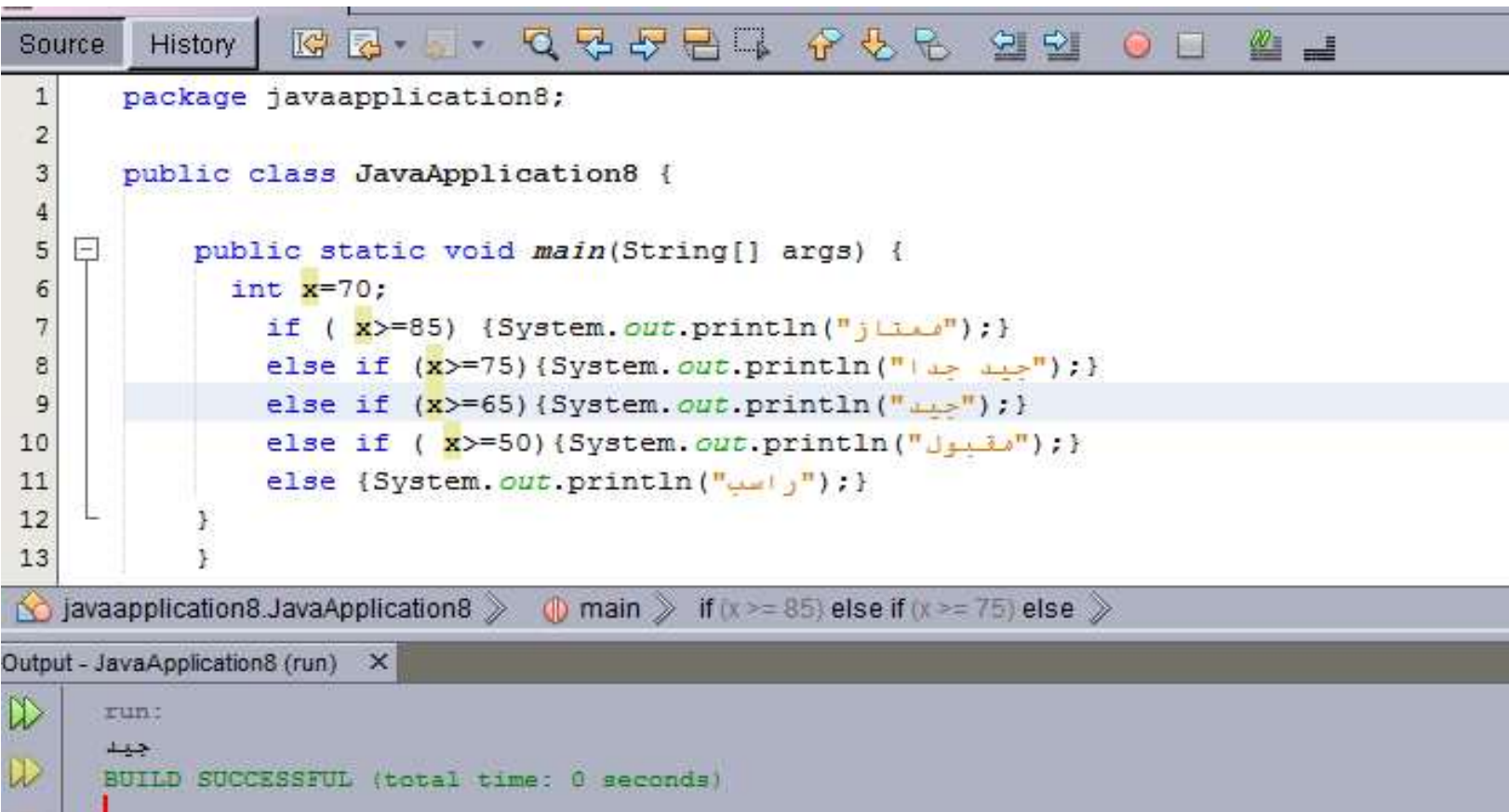
The screenshot shows an IDE window with a source code editor and an output console. The source code is a Java application named JavaApplication8. It contains a main method where a variable x is set to 50. An if statement checks if x is greater than or equal to 50. If true, it prints "ناجح" (Najih) to the console. If false, it prints "راسب" (Rasib). The output console shows the result of running the application, which is "ناجح". The build was successful.

```
1 package javaapplication8;
2
3 public class JavaApplication8 {
4
5     public static void main(String[] args) {
6         int x=50;
7         if (x>=50) {System.out.println("ناجح");}
8         else {System.out.println("راسب");}
9     }
10 }
11
```

Output - JavaApplication8 (run) X

run:
ناجح
BUILD SUCCESSFUL (total time: 0 seconds)

8- كود لطباعه تقدير الطالب بأستخدام if else



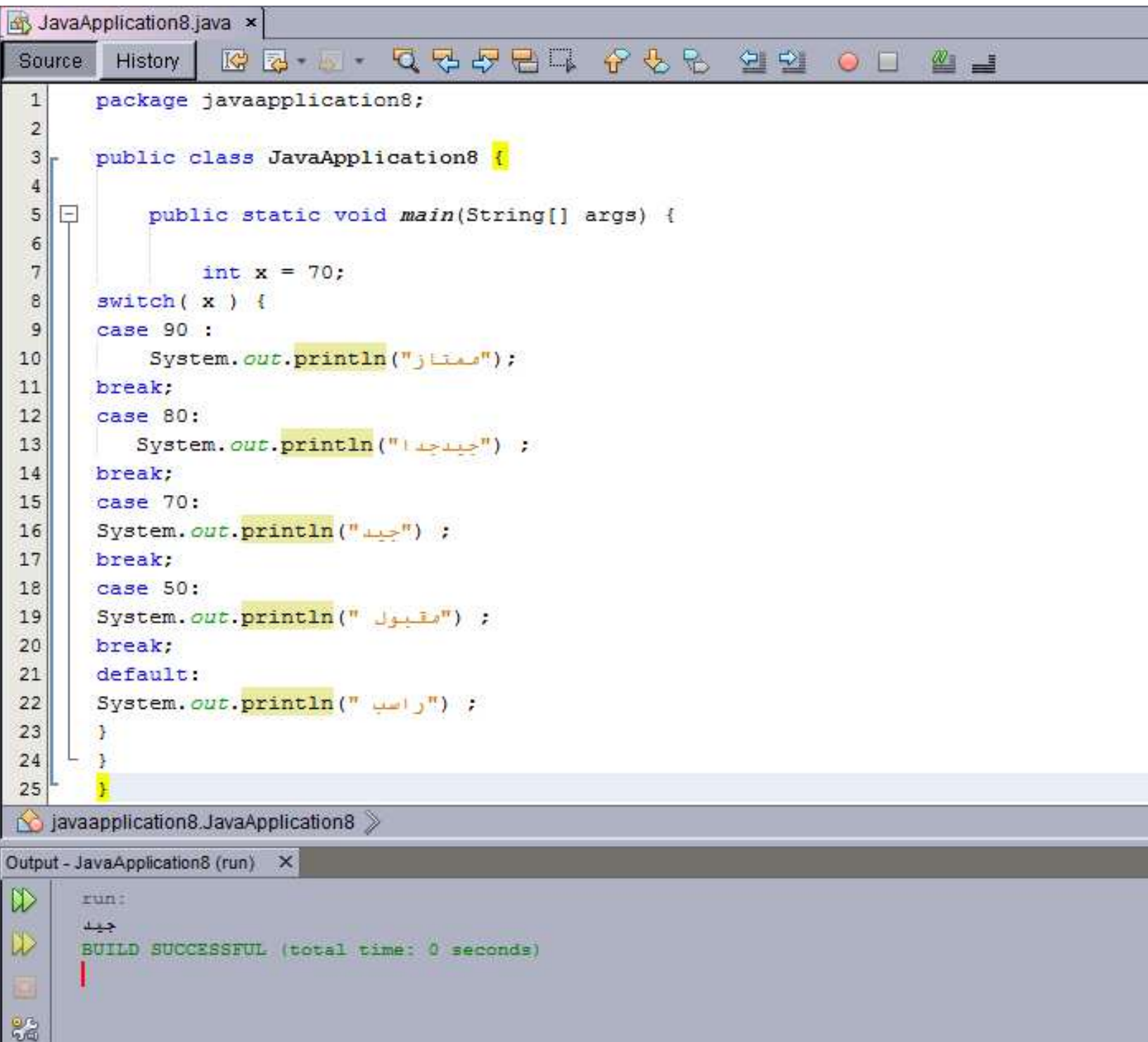
```
1 package javaapplication8;
2
3 public class JavaApplication8 {
4
5     public static void main(String[] args) {
6         int x=70;
7         if ( x>=85) {System.out.println("ممتاز");}
8         else if (x>=75) {System.out.println("جيد جدا");}
9         else if (x>=65) {System.out.println("جيد");}
10        else if ( x>=50) {System.out.println("مقبول");}
11        else {System.out.println("راسب");}
12    }
13 }
```

javaapplication8.JavaApplication8 > main > if (x >= 85) else if (x >= 75) else >

Output - JavaApplication8 (run) X

run:
جيد
BUILD SUCCESSFUL (total time: 0 seconds)

9 – كود لطباعه تقدير الطالب بأستخدام switch case



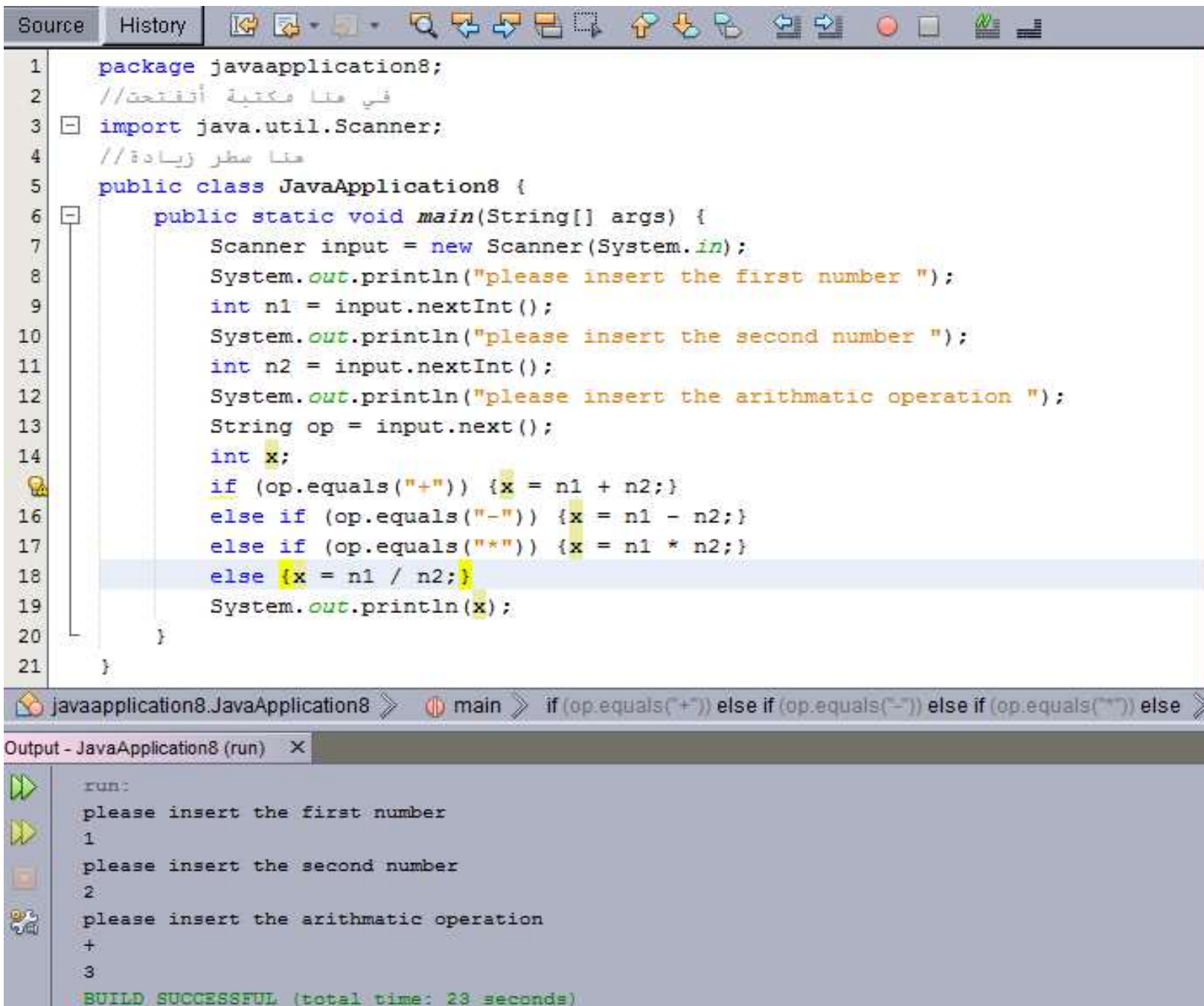
```
1 package javaapplication8;
2
3 public class JavaApplication8 {
4
5     public static void main(String[] args) {
6
7         int x = 70;
8         switch( x ) {
9             case 90 :
10                 System.out.println("ممتاز");
11                 break;
12             case 80:
13                 System.out.println("جيد جدا") ;
14                 break;
15             case 70:
16                 System.out.println("جيد") ;
17                 break;
18             case 50:
19                 System.out.println("مقبول") ;
20                 break;
21             default:
22                 System.out.println("راسب") ;
23             }
24         }
25     }
```

javaapplication8.JavaApplication8 >>

Output - JavaApplication8 (run) X

run:
جيد
BUILD SUCCESSFUL (total time: 0 seconds)

10- كود الاله الحاسبة بمدخلات من المستخدم



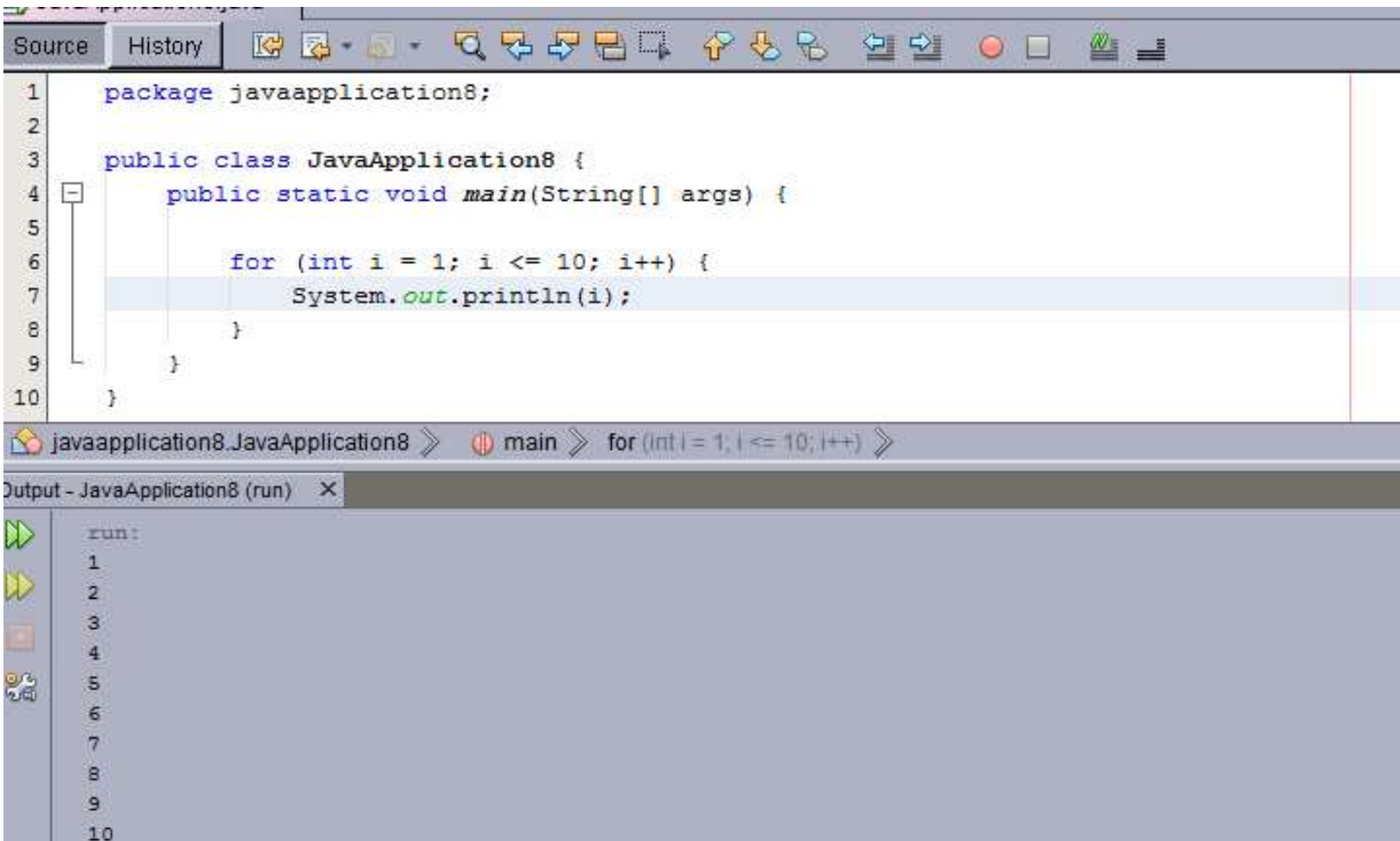
The screenshot displays an IDE with a Java source file named `JavaApplication8.java`. The code implements a simple calculator using the `Scanner` class for input. It prompts the user to enter two numbers and an arithmetic operation (+, -, *, /). The program then performs the calculation and prints the result. The output window shows the execution of the program with the inputs 1, 2, and '+', resulting in the output 3. The build was successful.

```
1 package javaapplication8;
2 // هنا مكتبة أتفتحت
3 import java.util.Scanner;
4 // هنا سطر زيادة
5 public class JavaApplication8 {
6     public static void main(String[] args) {
7         Scanner input = new Scanner(System.in);
8         System.out.println("please insert the first number ");
9         int n1 = input.nextInt();
10        System.out.println("please insert the second number ");
11        int n2 = input.nextInt();
12        System.out.println("please insert the arithmetic operation ");
13        String op = input.next();
14        int x;
15        if (op.equals("+")) {x = n1 + n2;}
16        else if (op.equals("-")) {x = n1 - n2;}
17        else if (op.equals("*")) {x = n1 * n2;}
18        else {x = n1 / n2;}
19        System.out.println(x);
20    }
21 }
```

Output - JavaApplication8 (run) X

```
run:
please insert the first number
1
please insert the second number
2
please insert the arithmetic operation
+
3
BUILD SUCCESSFUL (total time: 23 seconds)
```

11- كود التكرار for



The screenshot shows an IDE with a Java source file. The code defines a package, a class, and a main method with a for loop that prints numbers 1 through 10. The output window below shows the execution results, displaying the numbers 1 to 10 on separate lines.

```
1 package javaapplication8;
2
3 public class JavaApplication8 {
4     public static void main(String[] args) {
5
6         for (int i = 1; i <= 10; i++) {
7             System.out.println(i);
8         }
9     }
10 }
```

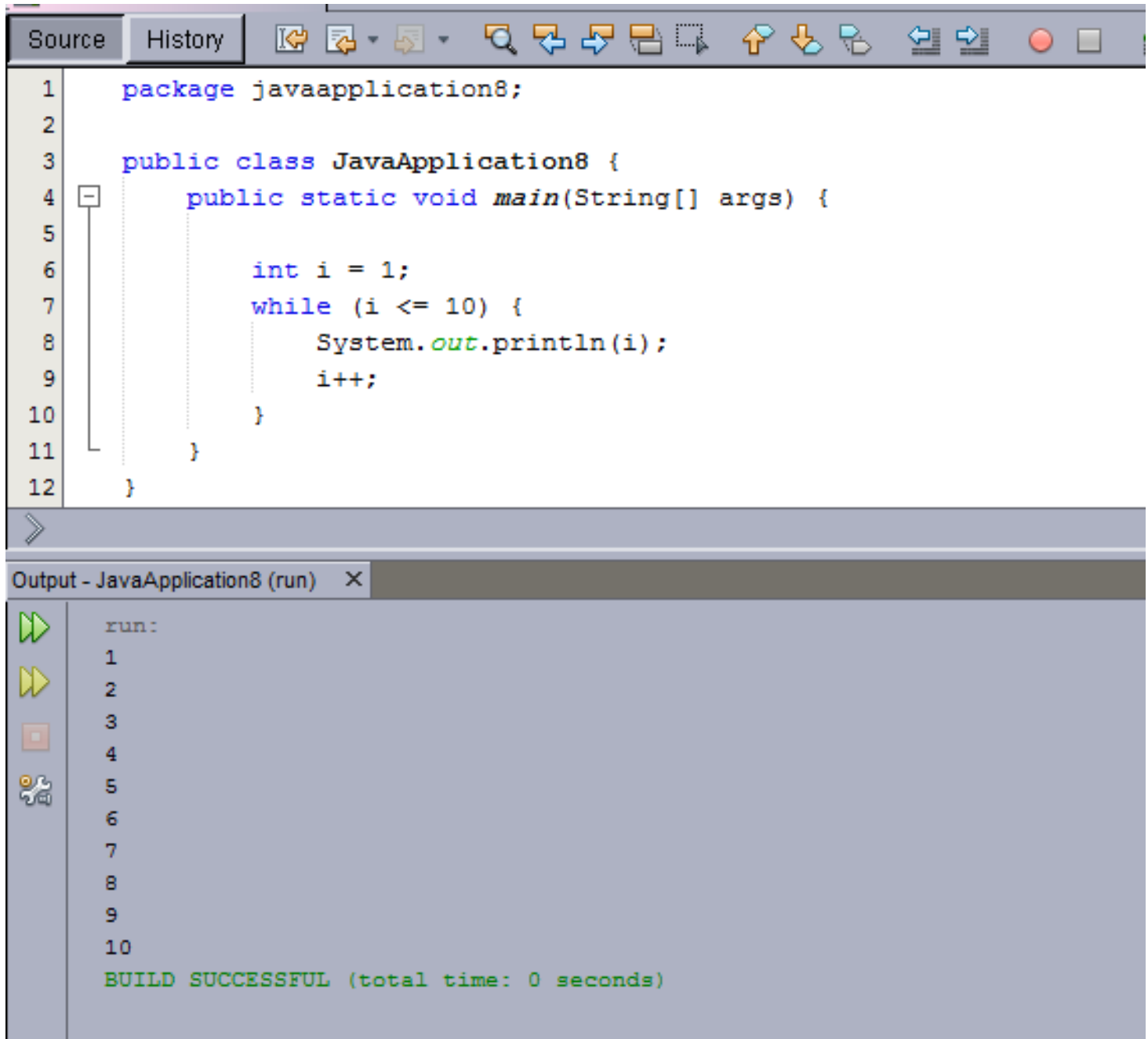
javaapplication8.JavaApplication8 > main > for (int i = 1; i <= 10; i++) >

Output - JavaApplication8 (run) X

run:

```
1
2
3
4
5
6
7
8
9
10
```

12 – كود التكرار while

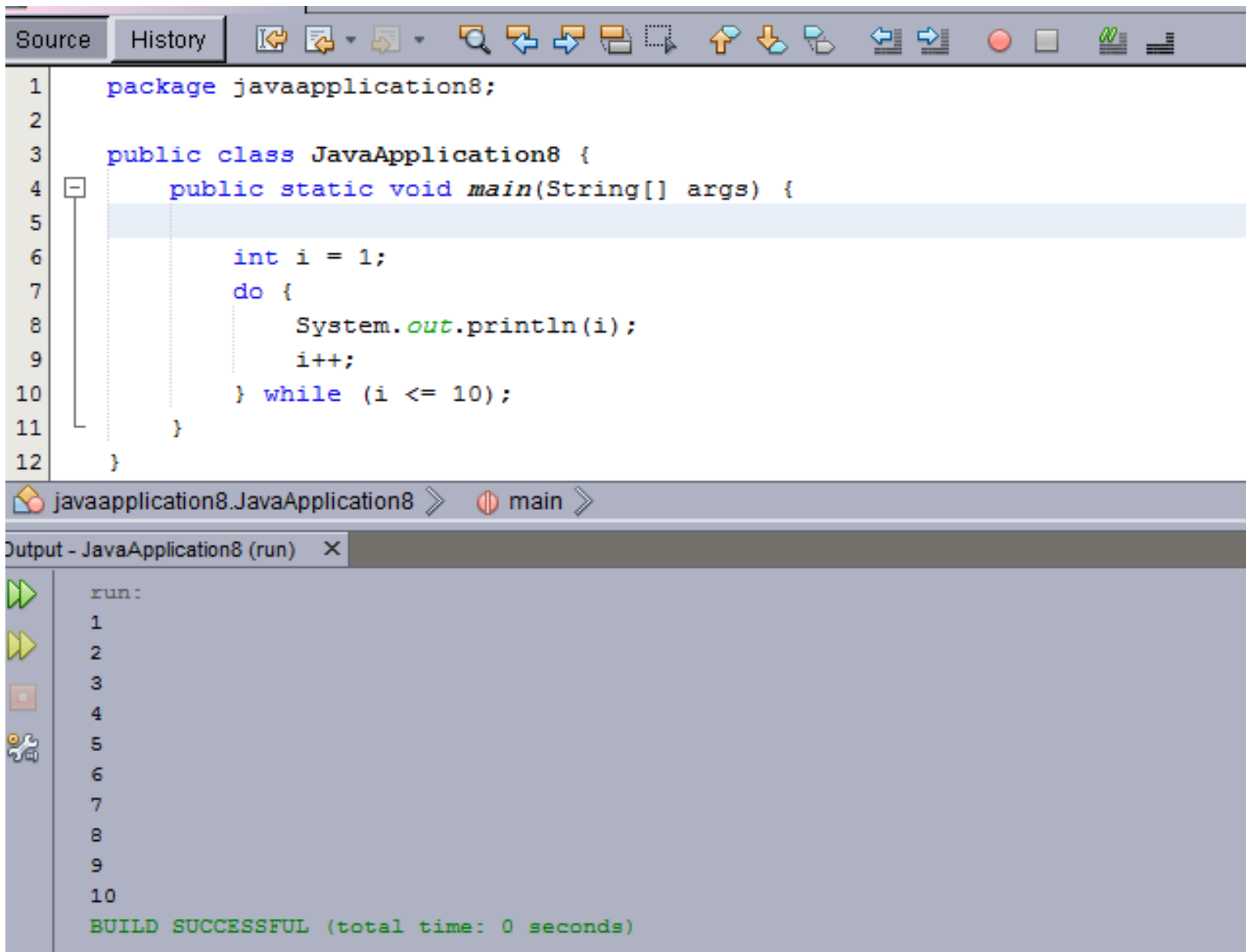


The screenshot shows an IDE window with two tabs: 'Source' and 'History'. The 'Source' tab is active, displaying a Java program. The code is as follows:

```
1 package javaapplication8;
2
3 public class JavaApplication8 {
4     public static void main(String[] args) {
5
6         int i = 1;
7         while (i <= 10) {
8             System.out.println(i);
9             i++;
10        }
11    }
12 }
```

Below the code editor is the 'Output - JavaApplication8 (run)' window. It shows the output of the program, which is the numbers 1 through 10, each on a new line. The output is preceded by 'run:'. At the bottom of the output window, it says 'BUILD SUCCESSFUL (total time: 0 seconds)'.

13- كود التكرار do while



The screenshot displays an IDE with a Java source file. The code defines a package `javaapplication8` and a public class `JavaApplication8`. Inside the class, the `main` method is implemented using a `do-while` loop to print numbers from 1 to 10. The loop initializes `i` to 1, prints it, increments it, and continues as long as `i` is less than or equal to 10. The IDE's output window shows the execution results, listing the numbers 1 through 10 and confirming a successful build.

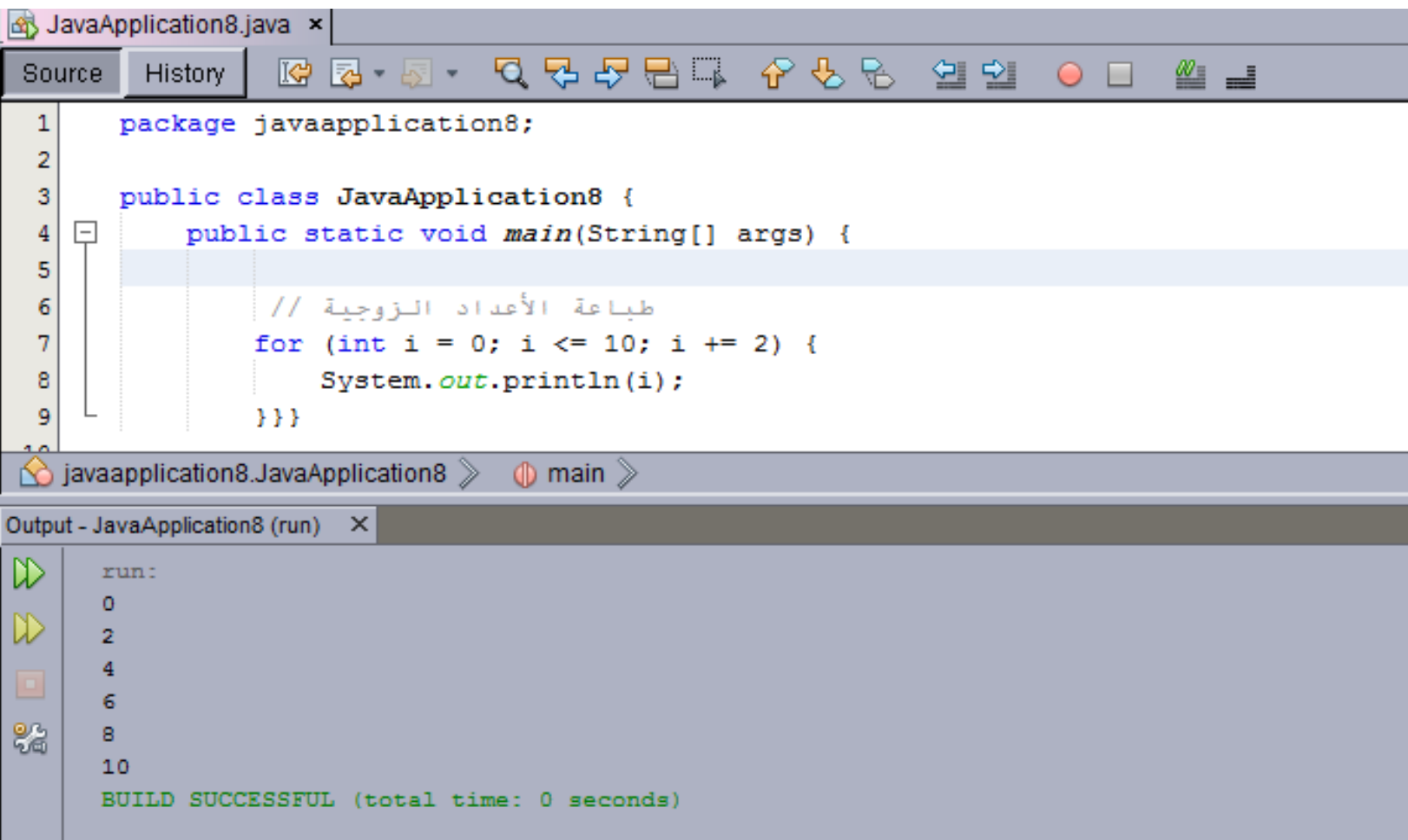
```
1 package javaapplication8;
2
3 public class JavaApplication8 {
4     public static void main(String[] args) {
5
6         int i = 1;
7         do {
8             System.out.println(i);
9             i++;
10        } while (i <= 10);
11    }
12 }
```

javaapplication8.JavaApplication8 > main >

Output - JavaApplication8 (run) X

run:
1
2
3
4
5
6
7
8
9
10
BUILD SUCCESSFUL (total time: 0 seconds)

14- كود طباعه الارقام الزوجية فقط



The screenshot shows an IDE window titled "JavaApplication8.java". The code is as follows:

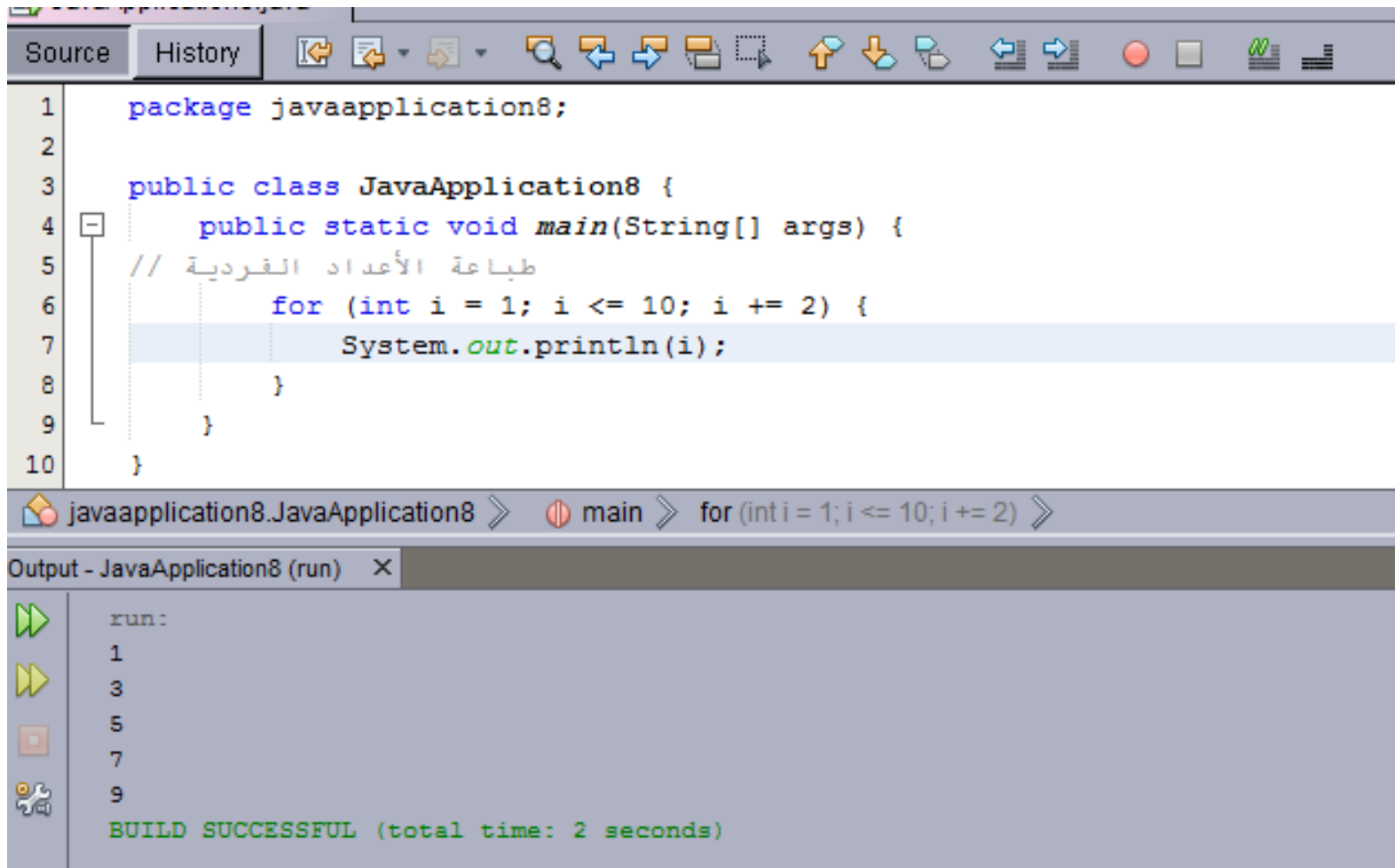
```
1 package javaapplication8;
2
3 public class JavaApplication8 {
4     public static void main(String[] args) {
5
6         // طباعة الأعداد الزوجية
7         for (int i = 0; i <= 10; i += 2) {
8             System.out.println(i);
9         }
10    }
```

The IDE's breadcrumb navigation shows the path: `javaapplication8.JavaApplication8` > `main`.

The "Output - JavaApplication8 (run)" window displays the following output:

```
run:
0
2
4
6
8
10
BUILD SUCCESSFUL (total time: 0 seconds)
```

15- كود طباعة الارقام الفردية فقط



The screenshot shows an IDE window with a Java source file. The code defines a package `javaapplication8` and a public class `JavaApplication8`. Inside the class, there is a `main` method that takes a `String[] args` parameter. A comment in Arabic indicates the purpose is to print odd numbers. A `for` loop is used to iterate from 1 to 10 in increments of 2, and `System.out.println(i)` is used to print each value. The IDE's breadcrumb navigation shows the current location: `javaapplication8.JavaApplication8 > main > for (int i = 1; i <= 10; i += 2)`. The output console at the bottom shows the results of running the program, displaying the odd numbers 1, 3, 5, 7, and 9, followed by a successful build message.

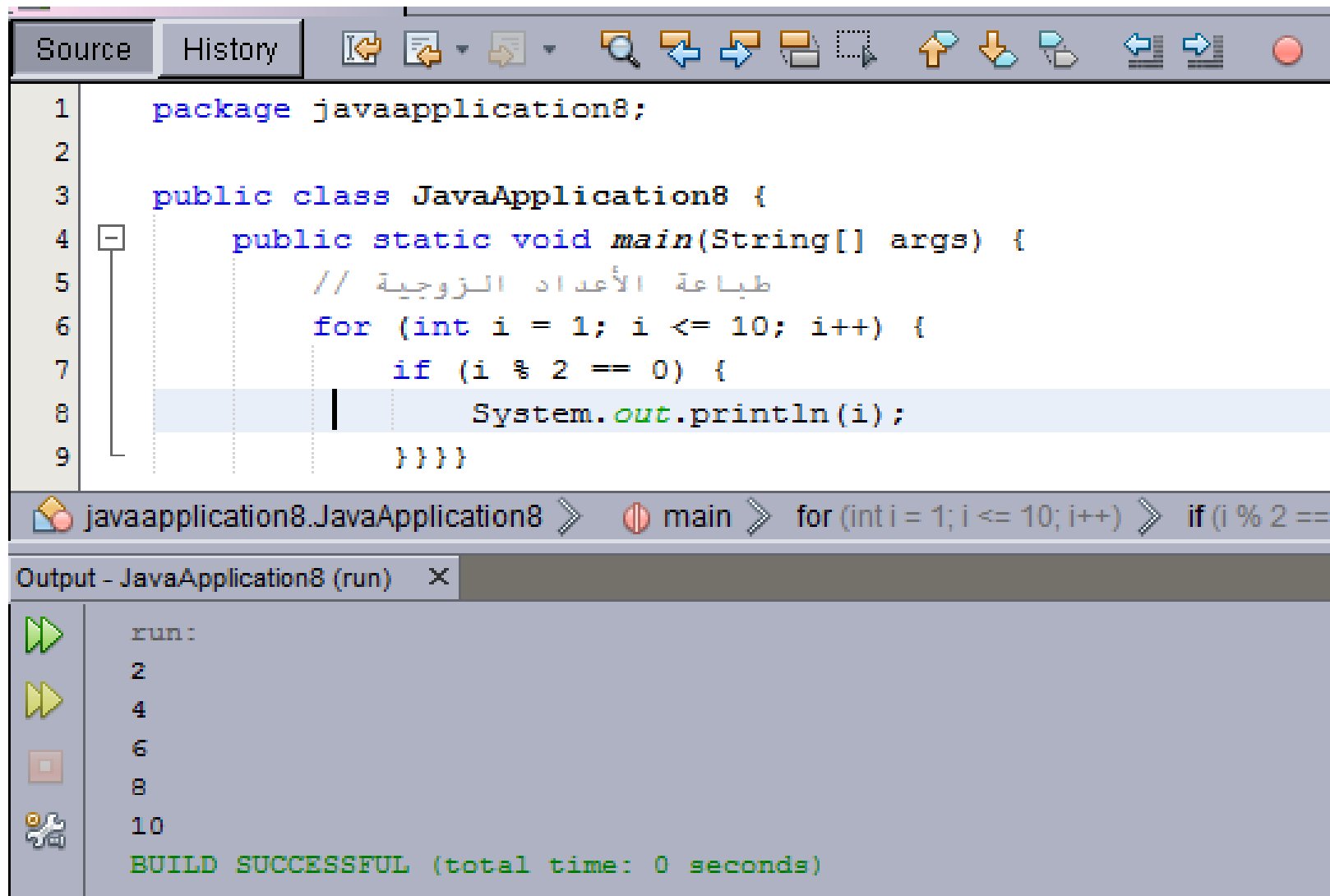
```
1 package javaapplication8;
2
3 public class JavaApplication8 {
4     public static void main(String[] args) {
5         // طباعة الأعداد الفردية
6         for (int i = 1; i <= 10; i += 2) {
7             System.out.println(i);
8         }
9     }
10 }
```

javaapplication8.JavaApplication8 > main > for (int i = 1; i <= 10; i += 2)

Output - JavaApplication8 (run) X

run:
1
3
5
7
9
BUILD SUCCESSFUL (total time: 2 seconds)

16- طباعه الارقام الزوجية فقط بأستخدام if



The screenshot shows an IDE with a Java source file. The code is as follows:

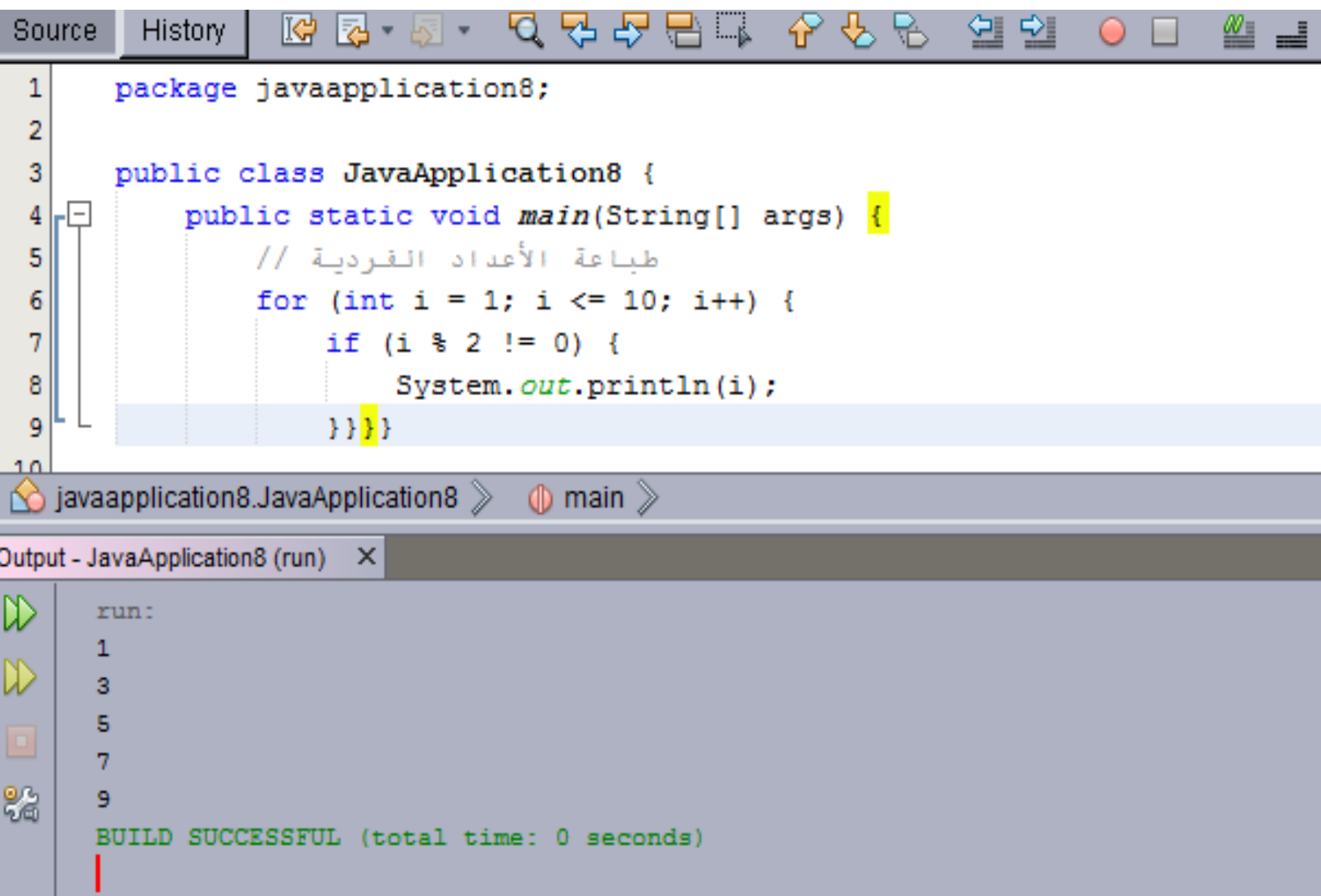
```
1 package javaapplication8;
2
3 public class JavaApplication8 {
4     public static void main(String[] args) {
5         // طباعة الأعداد الزوجية
6         for (int i = 1; i <= 10; i++) {
7             if (i % 2 == 0) {
8                 System.out.println(i);
9             }
10        }
11    }
12 }
```

The IDE's breadcrumb navigation shows the path: `javaapplication8.JavaApplication8 > main > for (int i = 1; i <= 10; i++) > if (i % 2 == 0)`.

The output window, titled "Output - JavaApplication8 (run)", displays the following:

```
run:
2
4
6
8
10
BUILD SUCCESSFUL (total time: 0 seconds)
```


17 - كود طباعة الارقام الفردية باستخدام if



The screenshot shows an IDE with a Java source file. The code is as follows:

```
1 package javaapplication8;
2
3 public class JavaApplication8 {
4     public static void main(String[] args) {
5         // طباعة الأعداد الفردية
6         for (int i = 1; i <= 10; i++) {
7             if (i % 2 != 0) {
8                 System.out.println(i);
9             }
10        }
11    }
12 }
```

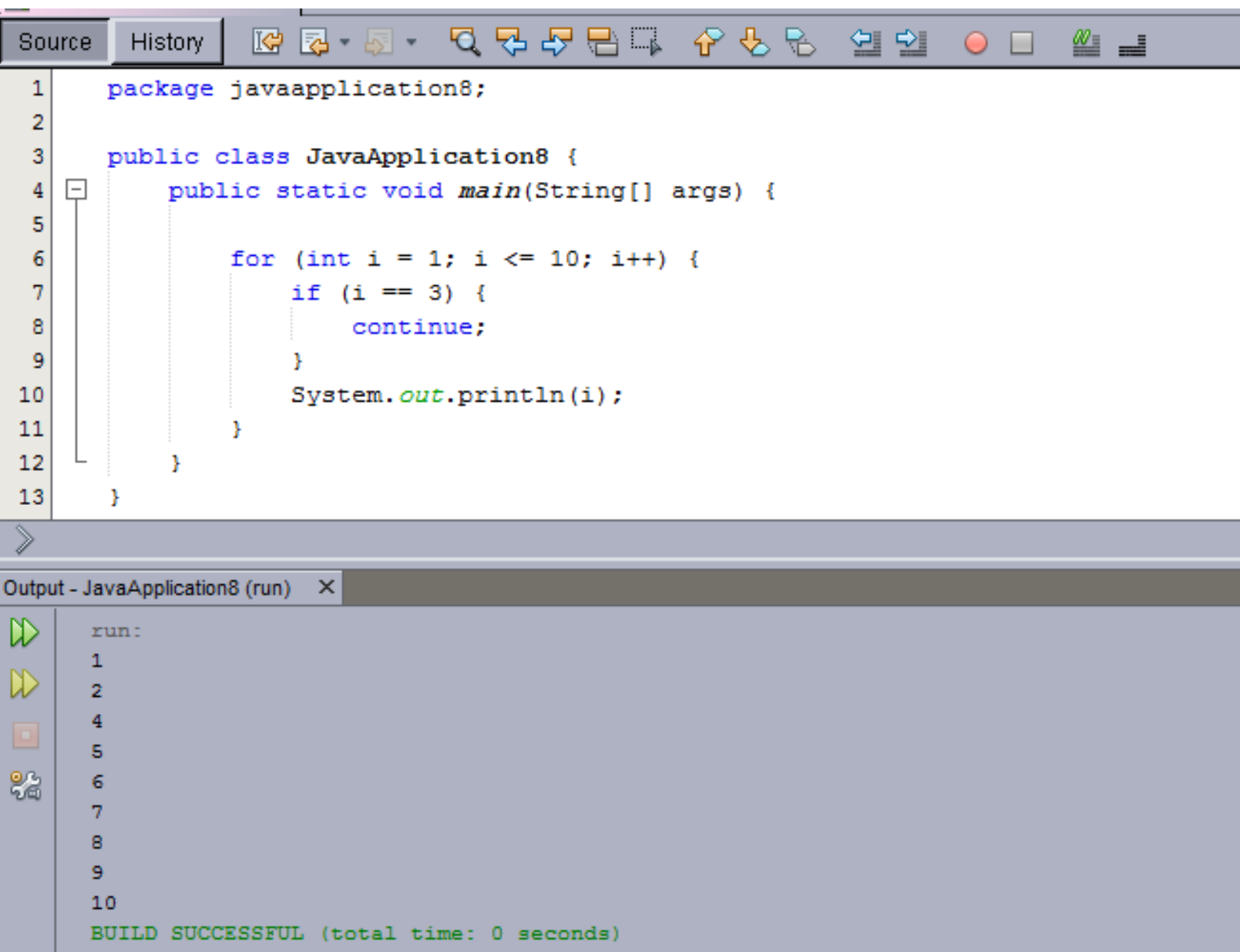
The IDE's breadcrumb shows the path: `javaapplication8.JavaApplication8 > main`.

The output window, titled "Output - JavaApplication8 (run)", shows the following:

```
run:
1
3
5
7
9
BUILD SUCCESSFUL (total time: 0 seconds)
```

The output displays the odd numbers 1, 3, 5, 7, and 9, each on a new line, followed by a successful build message.

18 – التكرار بدون طباعة العدد 3 مثلاً



The screenshot shows an IDE with a Java source file and its output. The source code is as follows:

```
1 package javaapplication8;
2
3 public class JavaApplication8 {
4     public static void main(String[] args) {
5
6         for (int i = 1; i <= 10; i++) {
7             if (i == 3) {
8                 continue;
9             }
10            System.out.println(i);
11        }
12    }
13 }
```

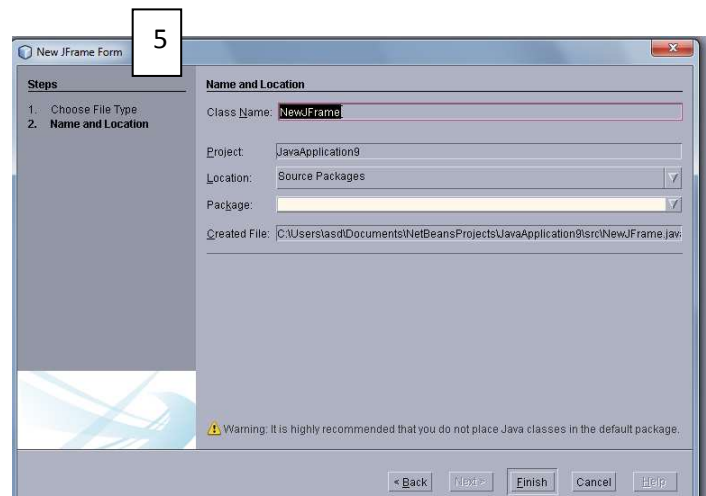
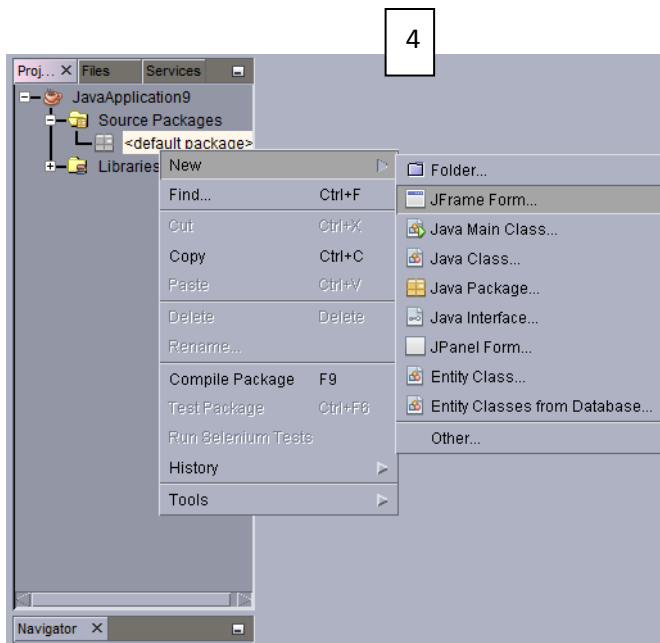
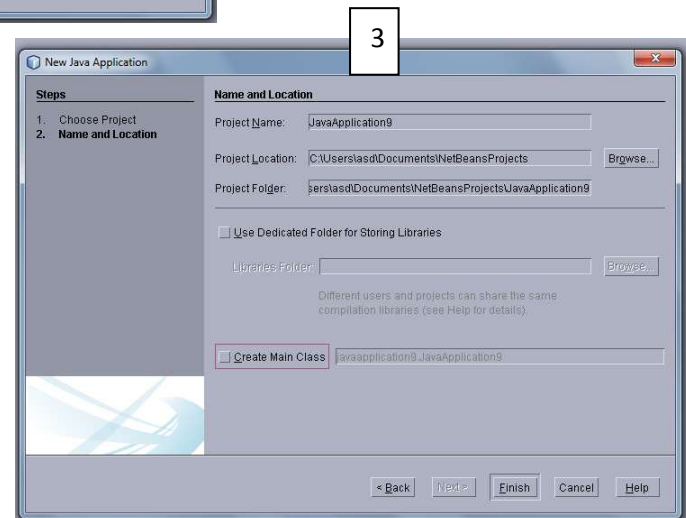
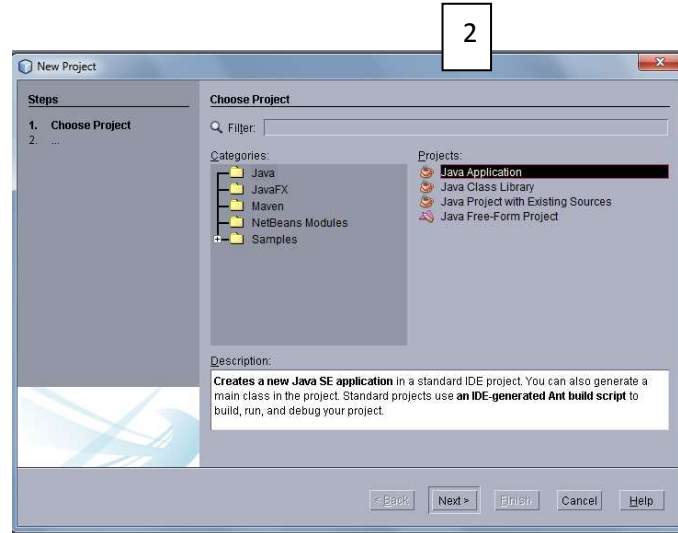
The output window, titled "Output - JavaApplication8 (run)", shows the following:

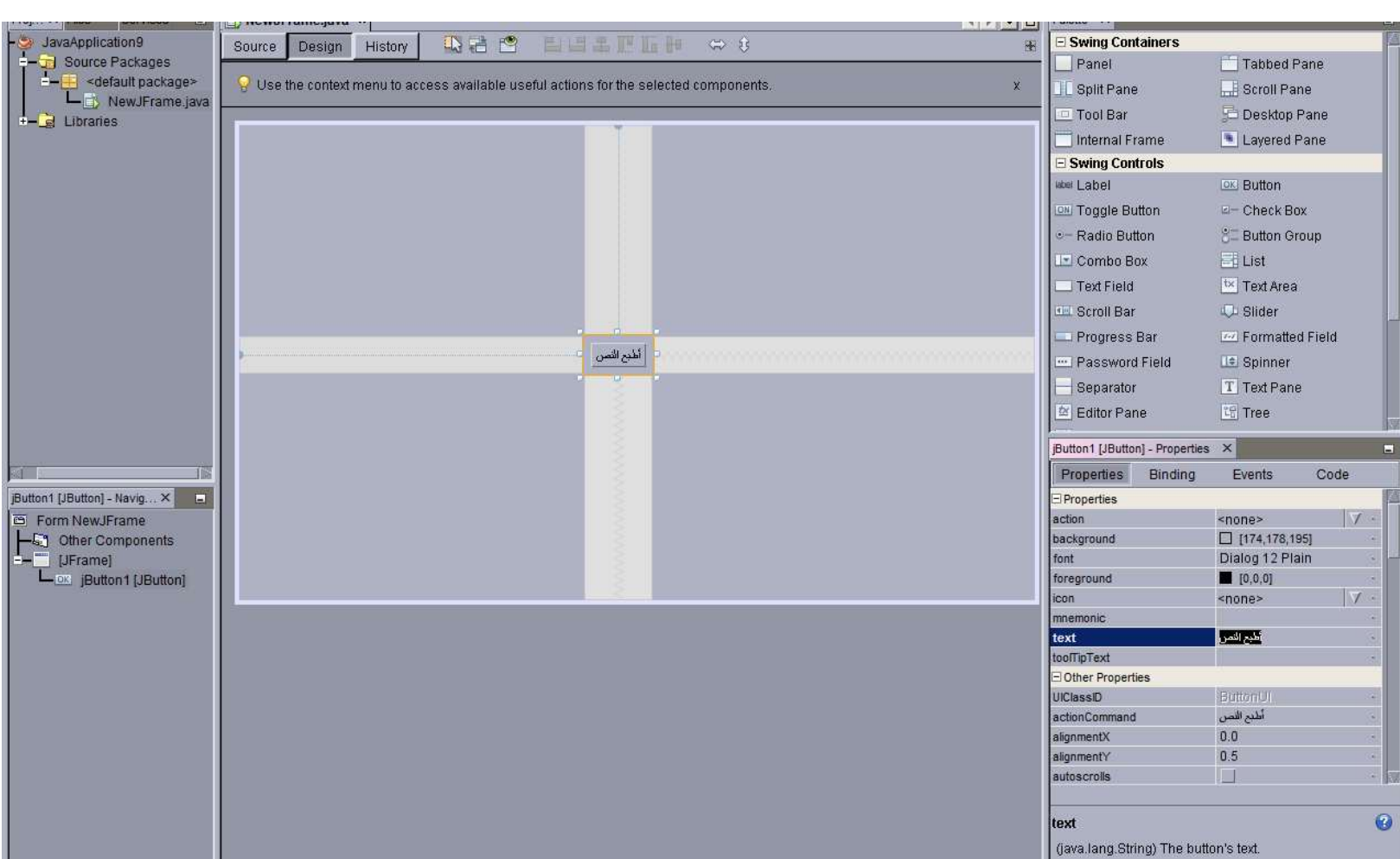
```
run:
1
2
4
5
6
7
8
9
10
BUILD SUCCESSFUL (total time: 0 seconds)
```

The output displays the numbers 1 through 10, with the number 3 omitted, demonstrating the effect of the `continue` statement.

بداية أول سكرشن
واجبة رسومية

طريقه فتح الواجهة الرسومية



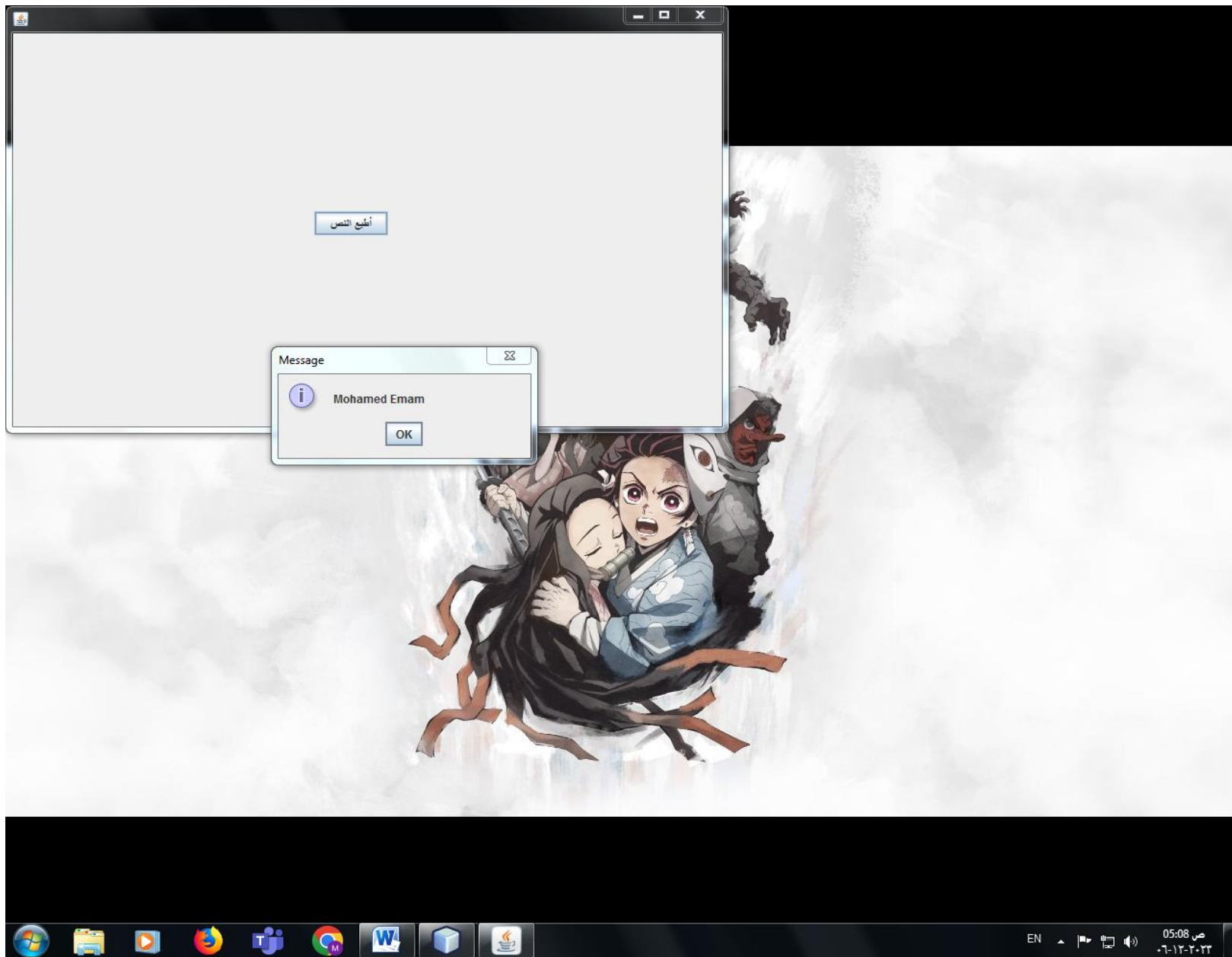


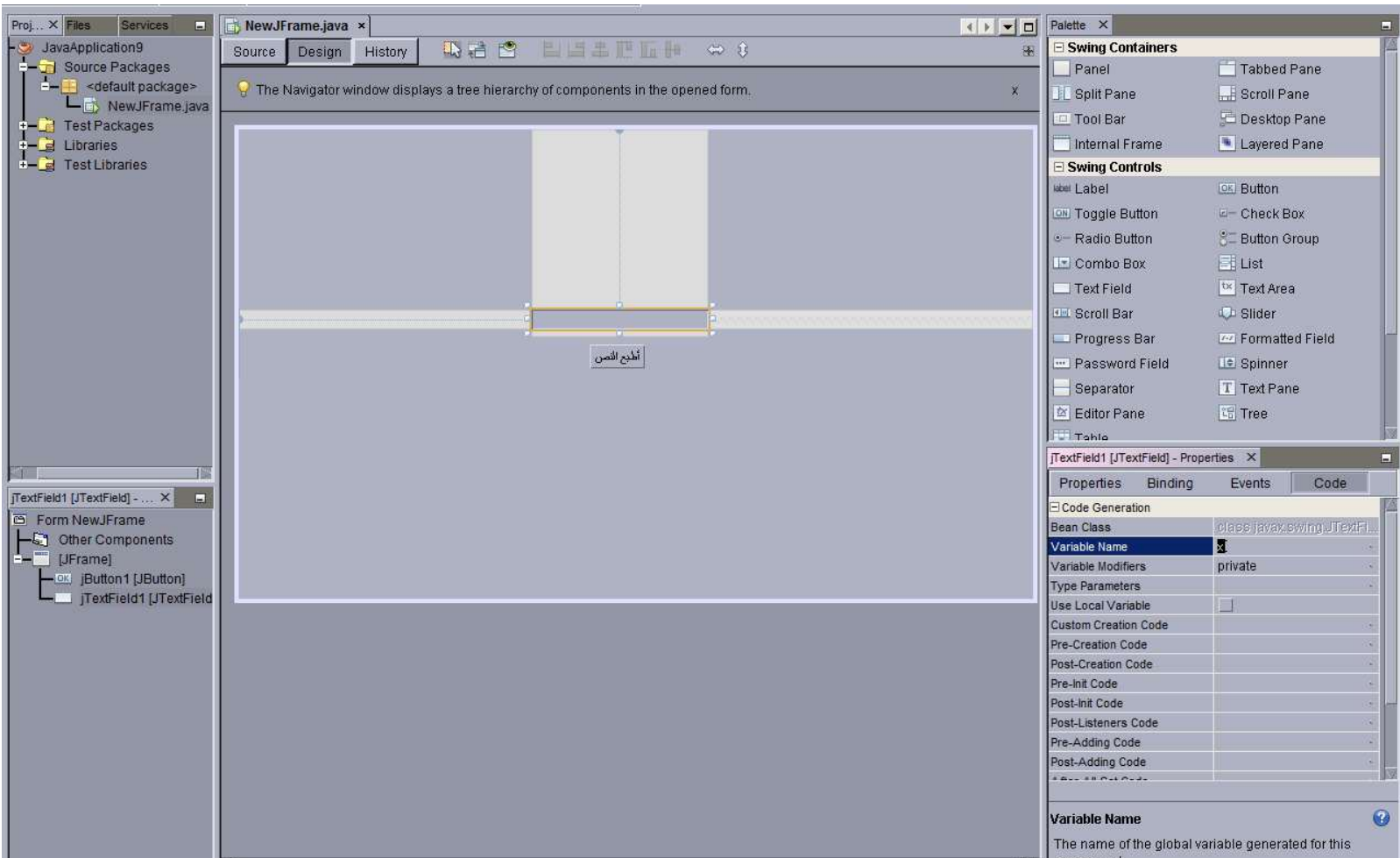
Source Design History

```

1 import javax.swing.JOptionPane;
2 // دا سطر مهم تمت اضافة
3 public class NewJFrame extends javax.swing.JFrame {
4
5
6     public NewJFrame() {
7         initComponents();
8     }
9
10
11     @SuppressWarnings("unchecked")
12     Generated Code
13
14
15
16     private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
17         JOptionPane.showMessageDialog(null, "Mohamed Eman");
18     }
19
20
21
22
23
24
25
26
27
28
29
30
31     public static void main(String args[]) {
32
33         java.awt.EventQueue.invokeLater(new Runnable() {
34             public void run() {
35                 new NewJFrame().setVisible(true);
36             }
37         });
38     }
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60 // Variables declaration - do not modify
61 private javax.swing.JButton jButton1;
62 // End of variables declaration
63
64

```

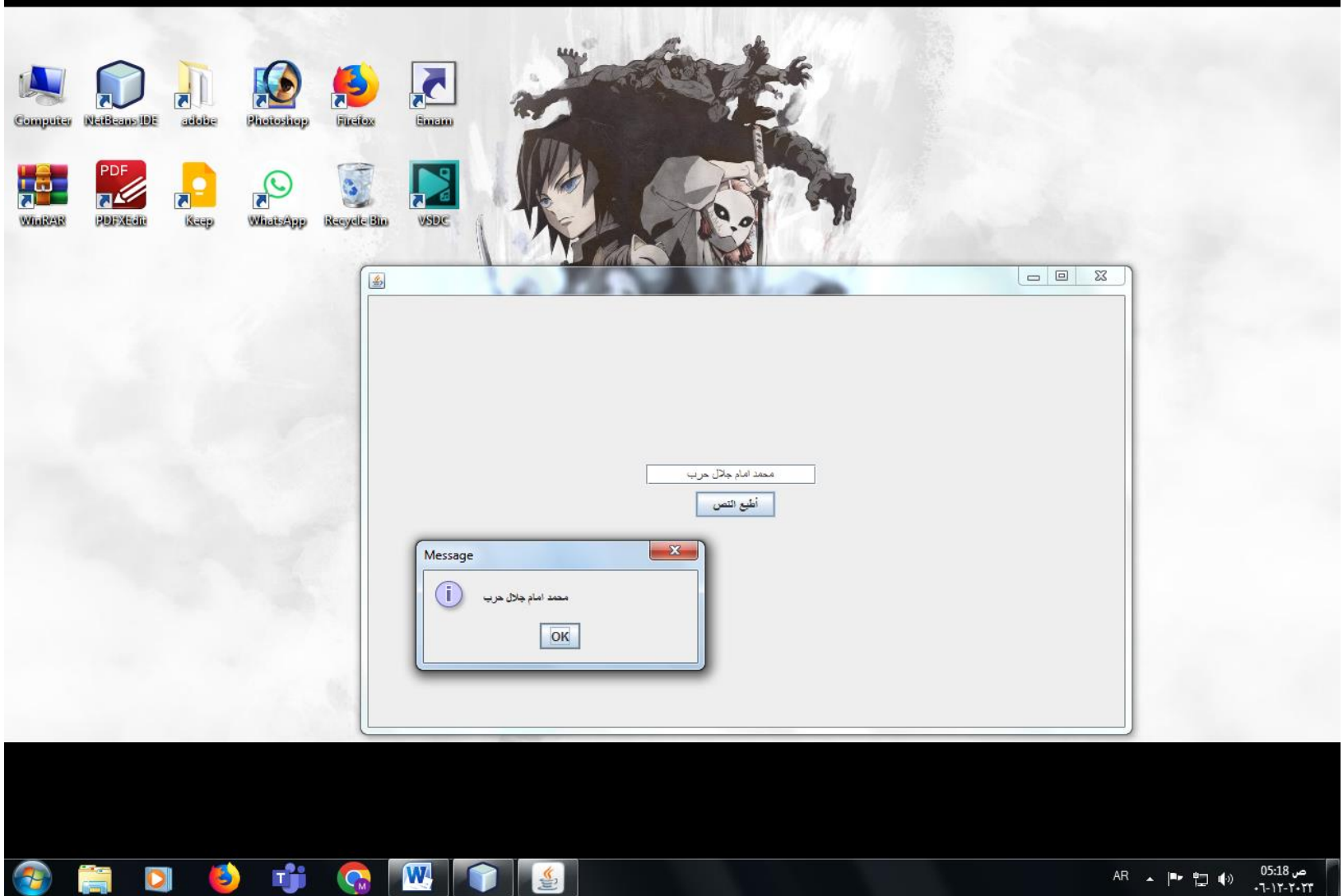




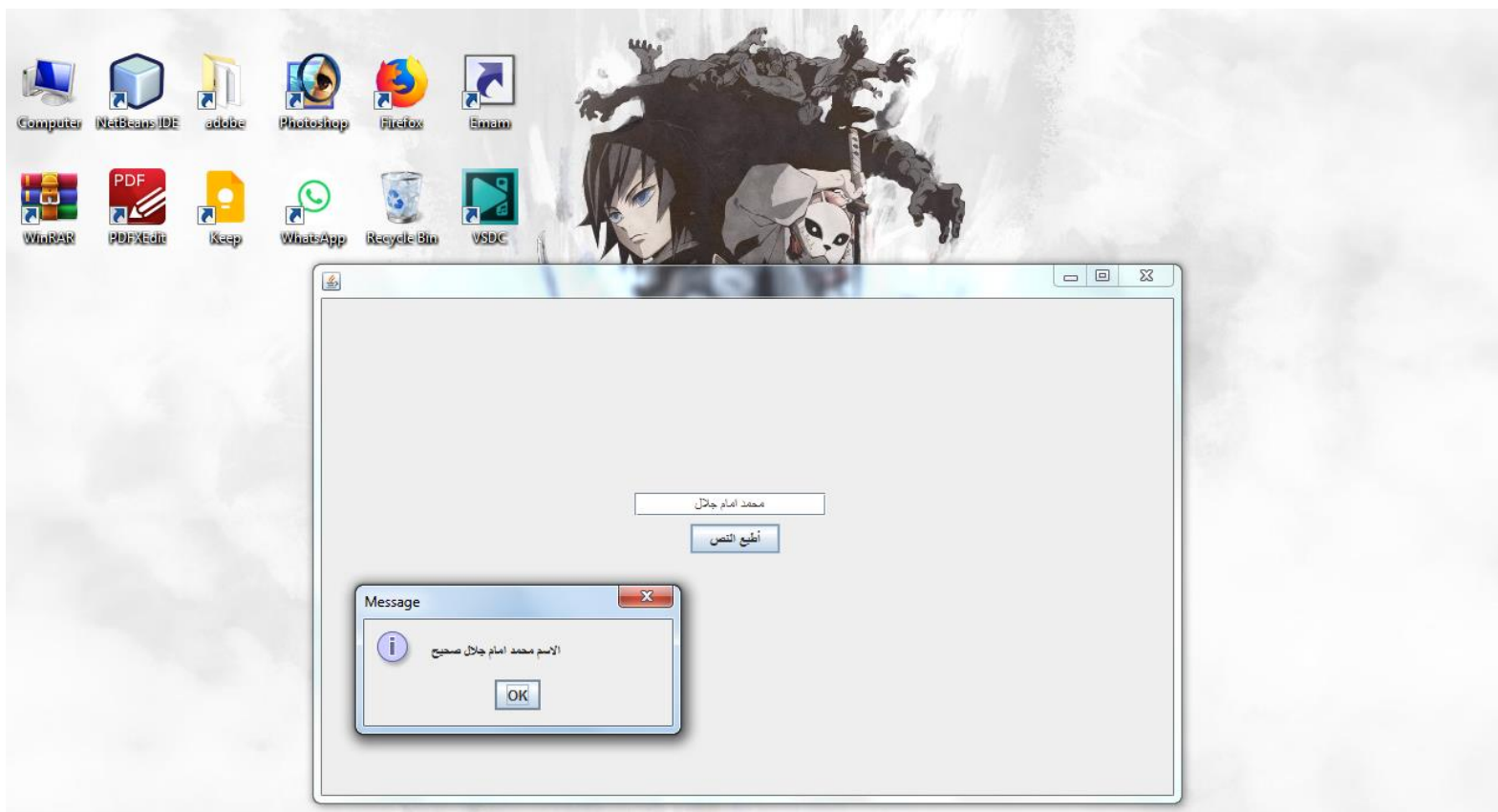
```

1  import javax.swing.JOptionPane;
2  // سطر مهم تمت إضافته
3  public class NewJFrame extends javax.swing.JFrame {
4
5
6      public NewJFrame() {
7          initComponents();
8      }
9
10
11     @SuppressWarnings("unchecked")
12     // <editor-fold defaultstate="collapsed" desc="Generated Code">
13     private void initComponents() { ...47 lines } // </editor-fold>
14
15
16     private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
17         String text = x.getText();
18         JOptionPane.showMessageDialog(null, text);
19     }
20
21
22     private void xActionPerformed(java.awt.event.ActionEvent evt) {
23         // TODO add your handling code here:
24     }
25
26
27     public static void main(String args[]) {
28
29         java.awt.EventQueue.invokeLater(new Runnable() {
30             public void run() {
31                 new NewJFrame().setVisible(true);
32             }
33         });
34     }
35
36     // Variables declaration - do not modify
37     private javax.swing.JButton jButton1;
38     private javax.swing.JTextField x;
39     // End of variables declaration
40 }

```

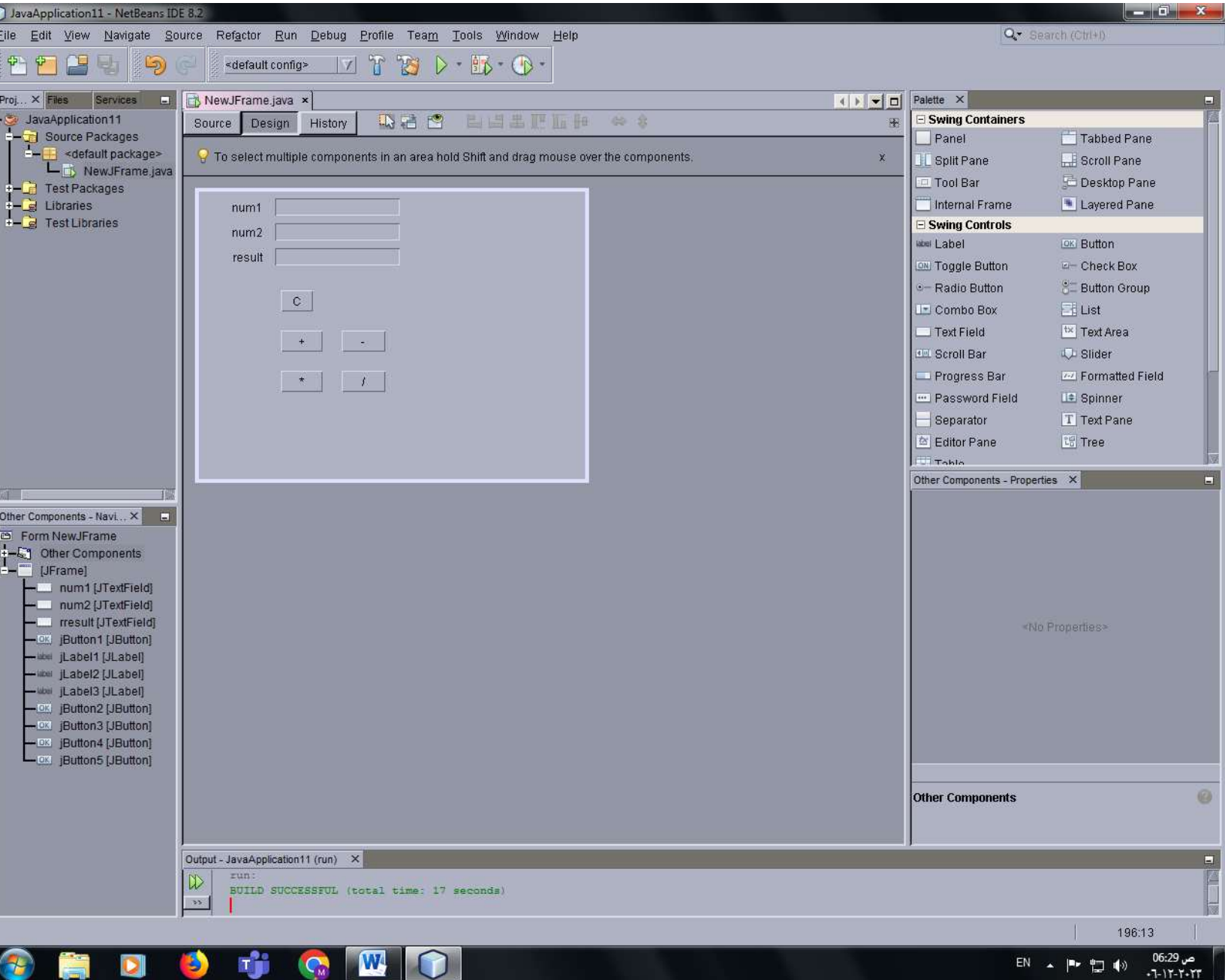




```
Source Design History
1 import javax.swing.JOptionPane;
2 // دا سطر مهم تمت إضافته
3 public class NewJFrame extends javax.swing.JFrame {
4
5
6     public NewJFrame() {
7         initComponents();
8     }
9
10
11     @SuppressWarnings("unchecked")
12     // <editor-fold defaultstate="collapsed" desc="Generated Code">
13     private void initComponents() {...47 lines} // </editor-fold>
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
62     String text = x.getText();
63     JOptionPane.showMessageDialog(null, "الاسم: " + text + " صحیح");
64 }
65
66 private void xActionPerformed(java.awt.event.ActionEvent evt) {
67     // TODO add your handling code here:
68 }
69
70
71 public static void main(String args[]) {
72     java.awt.EventQueue.invokeLater(new Runnable() {
73         public void run() {
74             new NewJFrame().setVisible(true);
75         }
76     });
77 }
78
79
80 // Variables declaration - do not modify
81 private javax.swing.JButton jButton1;
82 private javax.swing.JTextField x;
83 // End of variables declaration
84
85 }
```



.....كود الالة الحاسبة الكاملة.....

(1) قم بعمل الواجهة وخصصها من حيث الاسم و التنسيق كما تريد



(2) أكتب الكود المخصص لكل زر مع تغيير المعامل بما يتناسب + - *

```
154 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
155     double n1 = Double.parseDouble(num1.getText());
156     double n2 = Double.parseDouble(num2.getText());
157     double r=n1+n2;
158     rresult.setText(" "+r);
159     // TODO add your handling code here:
160 }
161
162 private void num1ActionPerformed(java.awt.event.ActionEvent evt) {...3 lines }
163
164 private void num2ActionPerformed(java.awt.event.ActionEvent evt) {...3 lines }
165
166 private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
167     num1.setText(" ");
168     num2.setText(" ");
169     rresult.setText(" ");
170     // TODO add your handling code here:
171 }
172
173 private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
174     double n1 = Double.parseDouble(num1.getText());
175     double n2 = Double.parseDouble(num2.getText());
176     double r=n1-n2;
177     rresult.setText(" "+r);
178     // TODO add your handling code here:
179 }
180
181 private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {
182     double n1 = Double.parseDouble(num1.getText());
183     double n2 = Double.parseDouble(num2.getText());
184     double r=n1/n2;
185     rresult.setText(" "+r);
186     // TODO add your handling code here:
187 }
188
189 private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
190     double n1 = Double.parseDouble(num1.getText());
191     double n2 = Double.parseDouble(num2.getText());
192     double r=n1*n2;
193     rresult.setText(" "+r);
194     // TODO add your handling code here:
195 }
196
197 ...
```

num1 16

num2 5

result 3.2

C

+

-

*

/



