

JAVA FUNDAMENTALS COURSE

EXERCISE

RETURN KEYWORD

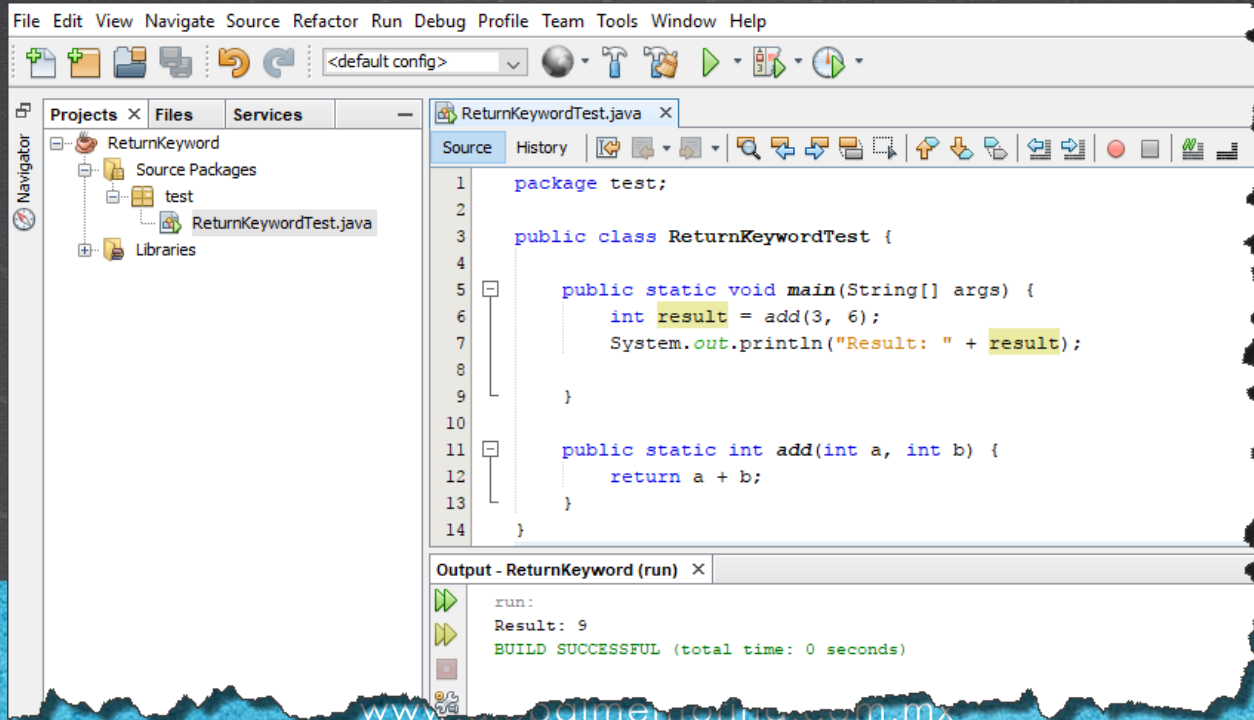


JAVA FUNDAMENTALS COURSE

www.globalmentoring.com.mx

EXERCISE OBJECTIVE

Create the exercise ReturnKeyword. At the end we should observe the following:



The screenshot displays an IDE interface with the following components:

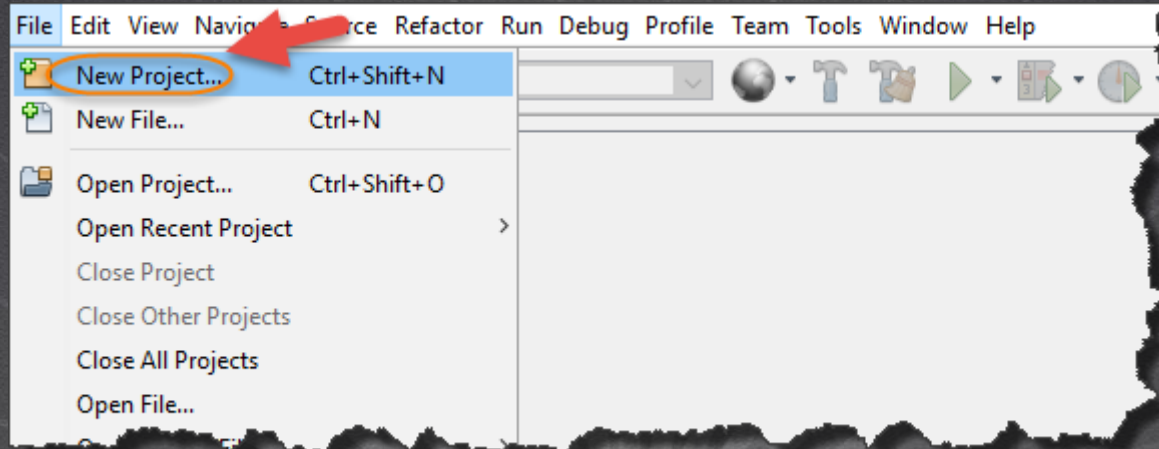
- Menu Bar:** File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help.
- Toolbar:** Includes icons for file operations, a configuration dropdown set to "<default config>", and execution buttons (run, debug, test).
- Navigator:** Shows a project named "ReturnKeyword" with a "test" package containing the file "ReturnKeywordTest.java".
- Source Editor:** Displays the code for "ReturnKeywordTest.java":

```
1 package test;  
2  
3 public class ReturnKeywordTest {  
4  
5     public static void main(String[] args) {  
6         int result = add(3, 6);  
7         System.out.println("Result: " + result);  
8     }  
9  
10  
11     public static int add(int a, int b) {  
12         return a + b;  
13     }  
14 }
```
- Output Console:** Shows the results of running the program:

```
run:  
Result: 9  
BUILD SUCCESSFUL (total time: 0 seconds)
```

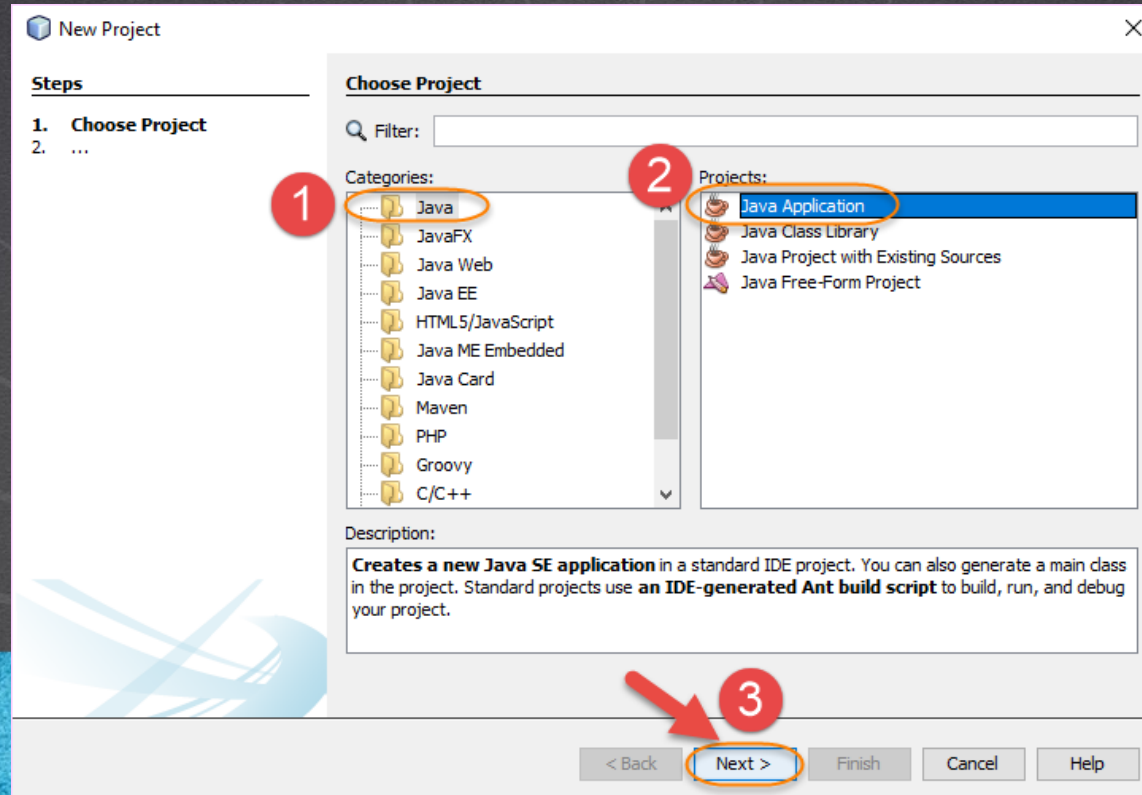

PASO 1. CREATE A NEW PROJECT

We are going to create the ReturnKeyword project.



PASO 1. CREATE A NEW PROJECT

Select Java -> Java Application and click on Next:



PASO 1. CREATE A NEW PROJECT

We are going to create the ReturnKeyword project.

New Java Application

Steps

1. Choose Project
2. **Name and Location**

Name and Location

Project Name:

Project Location:

Project Folder:

☐ Use Dedicated Folder for Storing Libraries

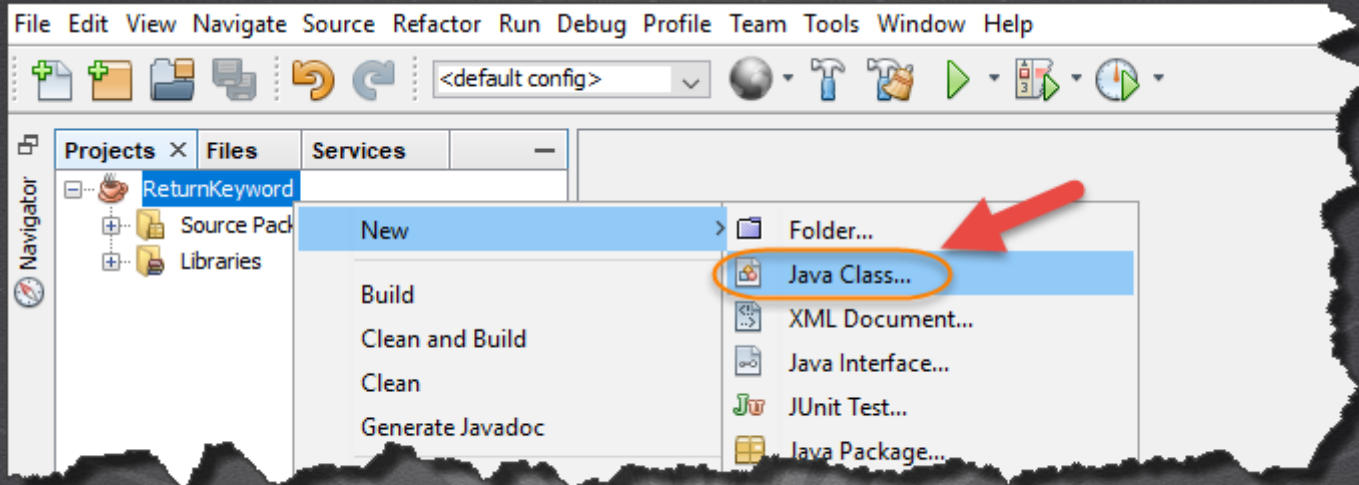
Libraries Folder:

Different users and projects can share the same compilation libraries (see Help for details).

☐ Create Main Class

PASO 2. CREATE A NEW CLASS

Create a new Java class:



JAVA FUNDAMENTALS COURSE

www.globalmentoring.com.mx

PASO 2. CREATE A NEW CLASS

Create a new Java class:

New Java Class

Steps

1. Choose File Type
2. **Name and Location**

Name and Location

Class Name: ReturnKeywordTest

Project: ReturnKeyword

Location: Source Packages

Package: test

Created File: C:\Courses\JavaFundamentals\Lesson13\ReturnKeyword\src\test\ReturnKeywordTest.java

< Back Next > **Finish** Cancel Help

JAVA FUNDAMENTALS COURSE

www.globalmentoring.com.mx

3. MODIFY THE CODE

ReturnKeywordTest.java:

```
package test;

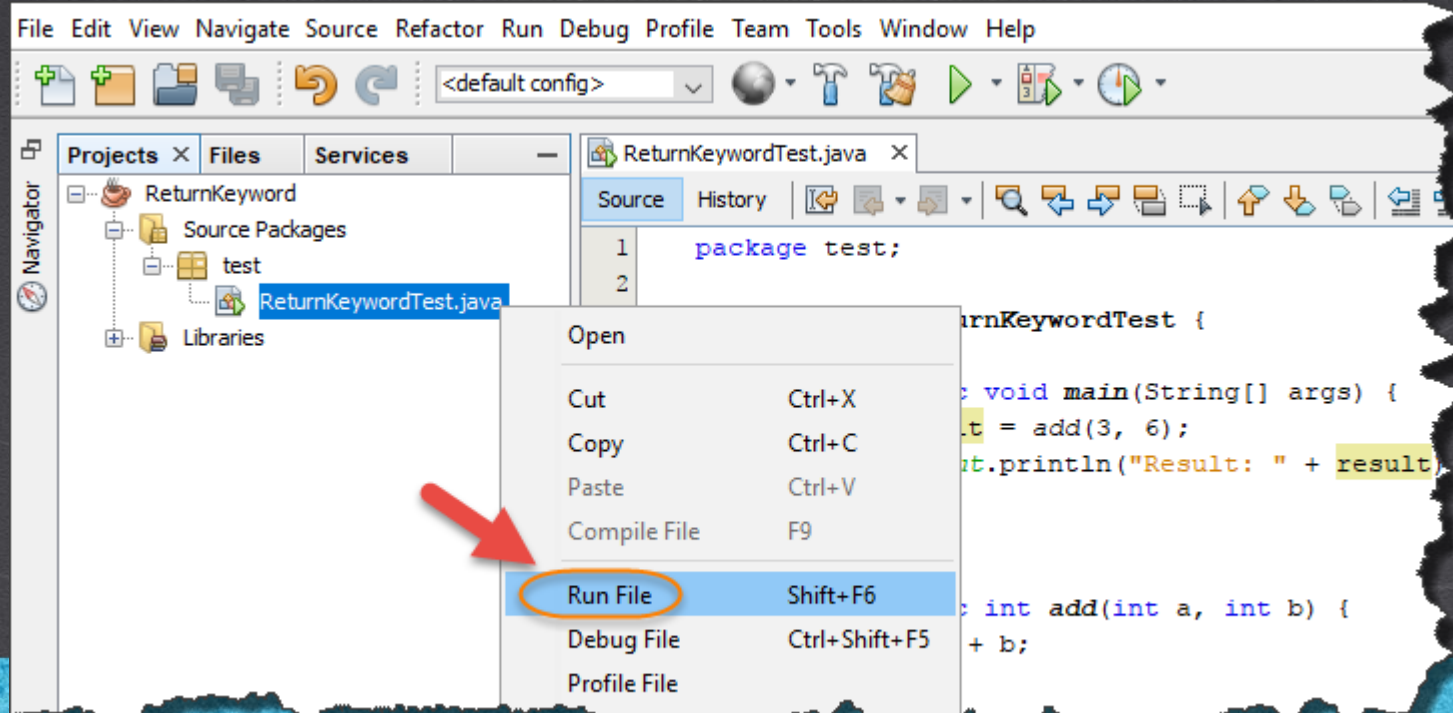
public class ReturnKeywordTest {

    public static void main(String[] args) {
        int result = add(3, 6);
        System.out.println("Result: " + result);
    }

    public static int add(int a, int b) {
        return a + b;
    }
}
```


4. EXECUTE THE PROJECT

Execute the project:

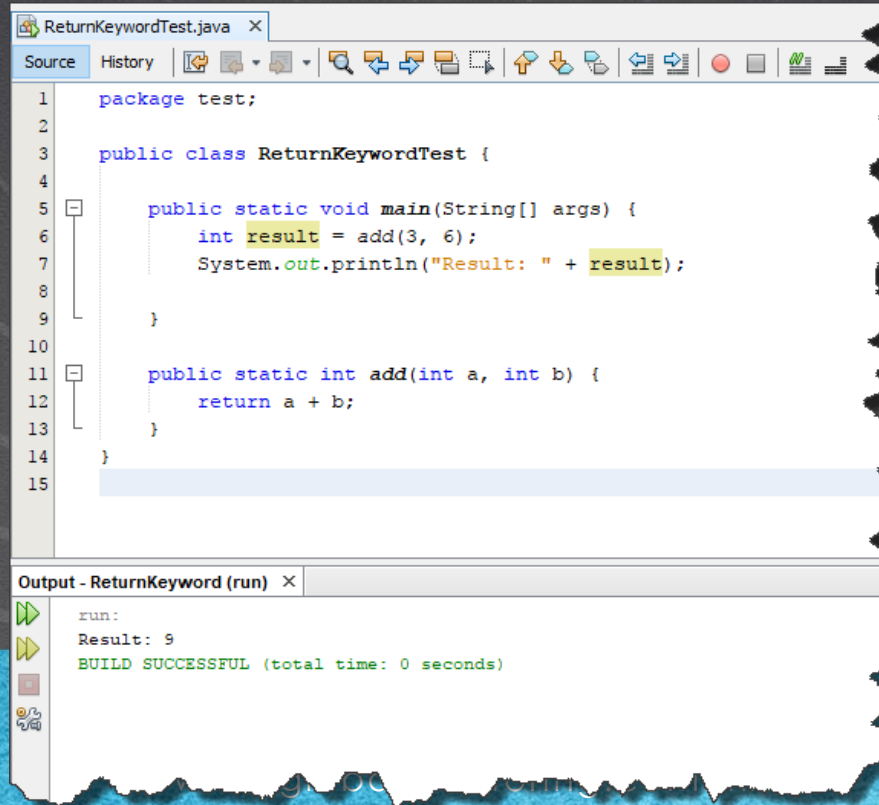


JAVA FUNDAMENTALS COURSE

www.globalmentoring.com.mx

4. EXECUTE THE PROJECT (CONT)

The result is as follows:



The screenshot shows an IDE window titled 'ReturnKeywordTest.java'. The code is as follows:

```
1 package test;
2
3 public class ReturnKeywordTest {
4
5     public static void main(String[] args) {
6         int result = add(3, 6);
7         System.out.println("Result: " + result);
8     }
9
10
11     public static int add(int a, int b) {
12         return a + b;
13     }
14 }
15
```

Below the code editor is the 'Output - ReturnKeyword (run)' window, which displays the following text:

```
run:
Result: 9
BUILD SUCCESSFUL (total time: 0 seconds)
```

5. MODIFY THE CODE

ReturnKeywordTest.java:

```
package test;

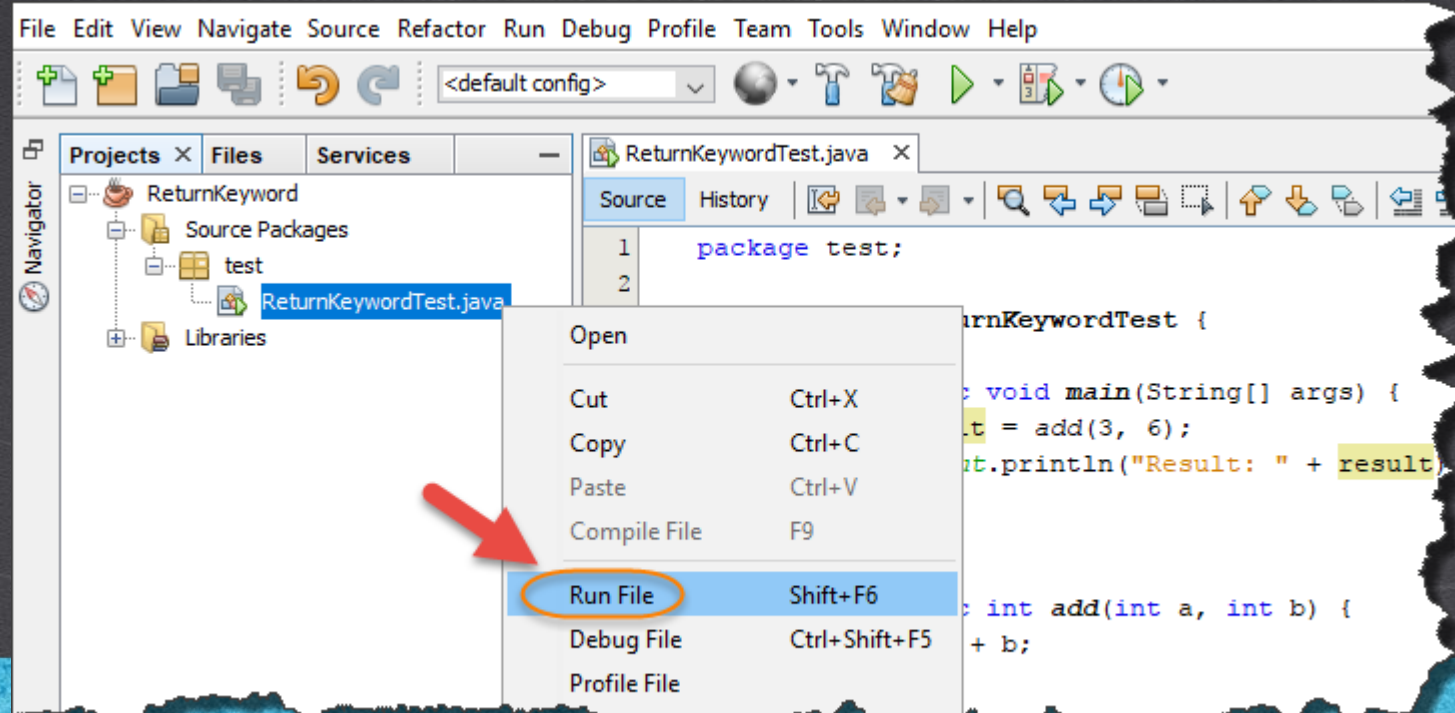
public class ReturnKeywordTest {

    public static void main(String[] args) {
        int result = add(0, 0);
        System.out.println("Result: " + result);
    }

    public static int add(int a, int b) {
        if (a == 0 && b == 0) {
            System.out.println("You must provide non-zero values");
            return 0;
        }
        return a + b;
    }
}
```


6. EXECUTE THE PROJECT

Execute the project:

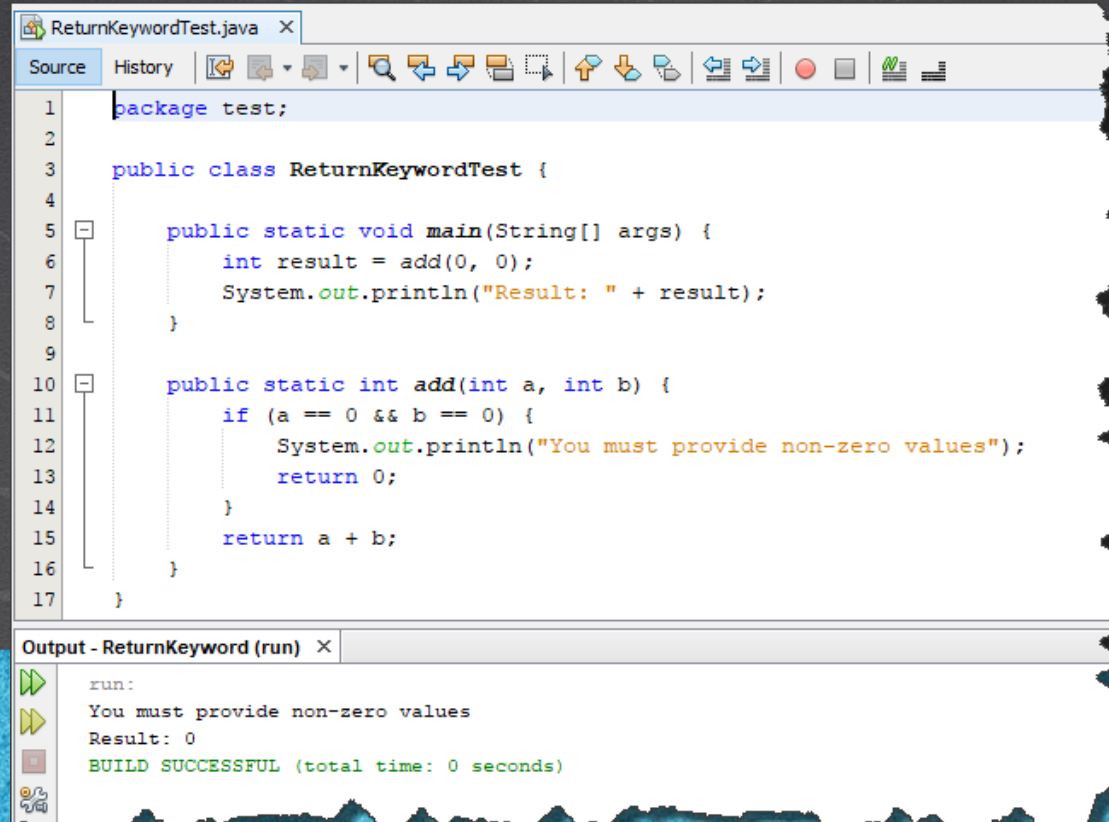


JAVA FUNDAMENTALS COURSE

www.globalmentoring.com.mx

6. EXECUTE THE PROJECT (CONT)

The result is as follows:



The screenshot displays an IDE window titled 'ReturnKeywordTest.java'. The code defines a package 'test' and a class 'ReturnKeywordTest' with two methods: 'main' and 'add'. The 'main' method calls 'add(0, 0)' and prints the result. The 'add' method checks if both arguments are zero; if so, it prints an error message and returns 0; otherwise, it returns the sum. Below the code editor, the 'Output - ReturnKeyword (run)' window shows the execution results: 'run:', 'You must provide non-zero values', 'Result: 0', and 'BUILD SUCCESSFUL (total time: 0 seconds)'.

```
1 package test;
2
3 public class ReturnKeywordTest {
4
5     public static void main(String[] args) {
6         int result = add(0, 0);
7         System.out.println("Result: " + result);
8     }
9
10    public static int add(int a, int b) {
11        if (a == 0 && b == 0) {
12            System.out.println("You must provide non-zero values");
13            return 0;
14        }
15        return a + b;
16    }
17 }
```

run:
You must provide non-zero values
Result: 0
BUILD SUCCESSFUL (total time: 0 seconds)

EXERCISE CONCLUSION

- With this exercise we have put into practice the use of the return keyword using the primitive types.
- We also saw how to use this word by adding conditioned code, in this case using an if block, however it can be any conditioned code.
- In the following exercise we will see an exercise to use the return keyword with Object types.

ONLINE COURSE

JAVA FUNDAMENTALS

By: Eng. Ubaldo Acosta



JAVA FUNDAMENTALS COURSE

www.globalmentoring.com.mx