

I/O File read/write

The image shows an IDE with two Java files open. The top file, `TestFileWriter.java`, is in the `CharStream` package and imports `java.io.FileWriter` and `java.io.IOException`. It contains a `main` method that writes the string "Welcome Bruh" to `output.txt` and then closes the `FileWriter`.

The bottom file, `TestFileReader.java`, is also in the `CharStream` package and imports `java.io.*`. It contains a `main` method that reads the file `output.txt` using a `FileReader`. It first reads the file character by character, printing each character, and then reads the entire file into a character array `cbuf` and prints it.

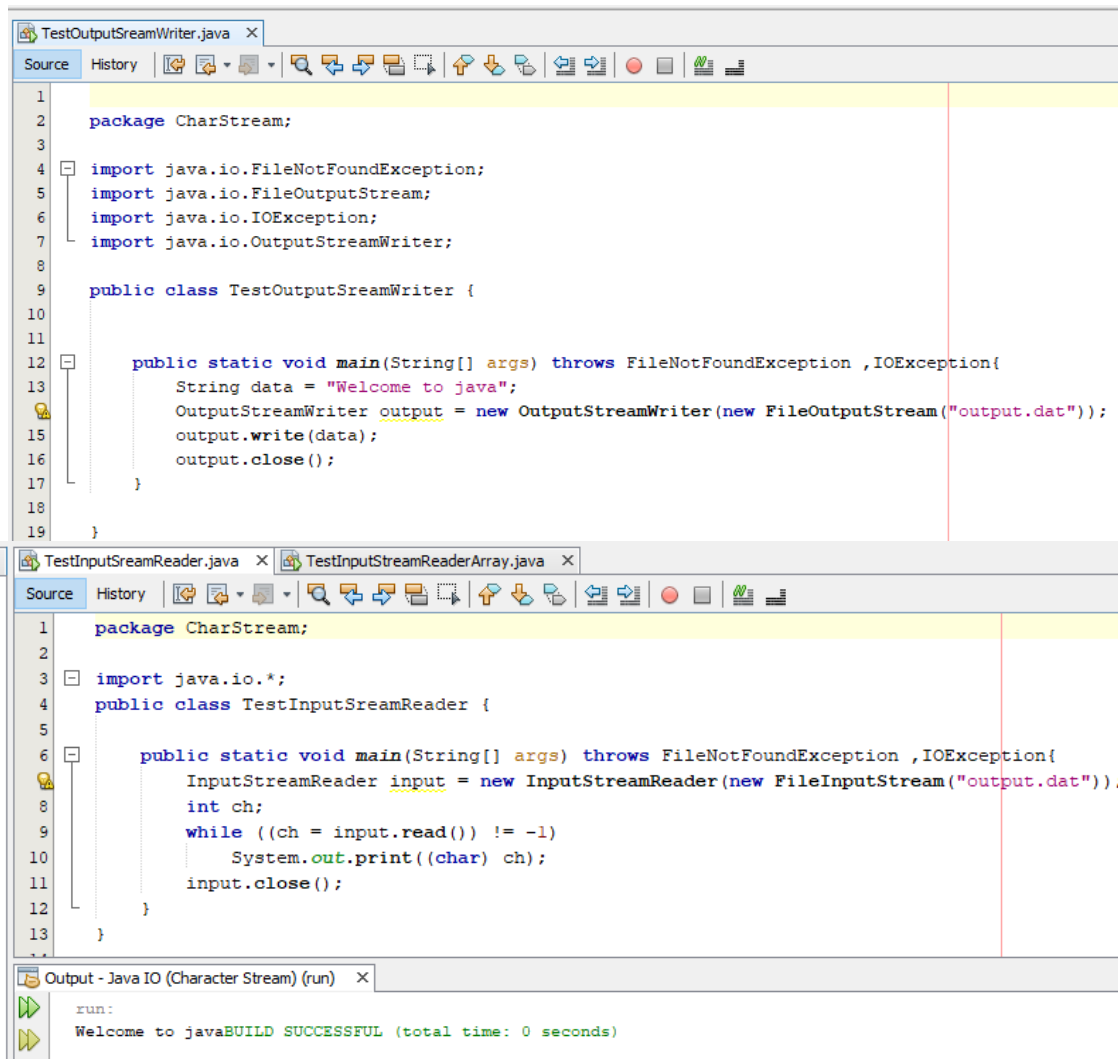
The output window at the bottom shows the results of running the program. It displays the text "Welcome Bruh" twice, indicating that the file was read correctly. The output window also shows "BUILD SUCCESSFUL (total time: 0 seconds)".

```
TestFileWriter.java
1 package CharStream;
2
3 import java.io.FileWriter;
4 import java.io.IOException;
5
6
7 public class TestFileWriter {
8     public static void main(String[] args) throws IOException {
9         String data = "Welcome Bruh";
10        FileWriter output = new FileWriter("output.txt");
11        output.write(data);
12        output.close();
13    }
14 }
15

TestFileReader.java
1 package CharStream;
2
3 import java.io.*;
4
5 public class TestFileReader {
6     public static void main(String[] args) throws FileNotFoundException, IOException {
7         //Reading a single character
8         FileReader input = new FileReader("output.txt");
9         int ch;
10        while ((ch = input.read()) != -1)
11            System.out.print((char) ch);
12        input.read();
13
14        System.out.println("");
15
16        //Reading characters into an array
17        char[] cbuf = new char[100];
18        FileReader in = new FileReader("output.txt");
19        in.read(cbuf);
20        System.out.println(cbuf);
21        in.close();
22    }
23 }
24

Output - Java IO (Character Stream) (run)
run:
Welcome Bruh
Welcome Bruh
BUILD SUCCESSFUL (total time: 0 seconds)
```

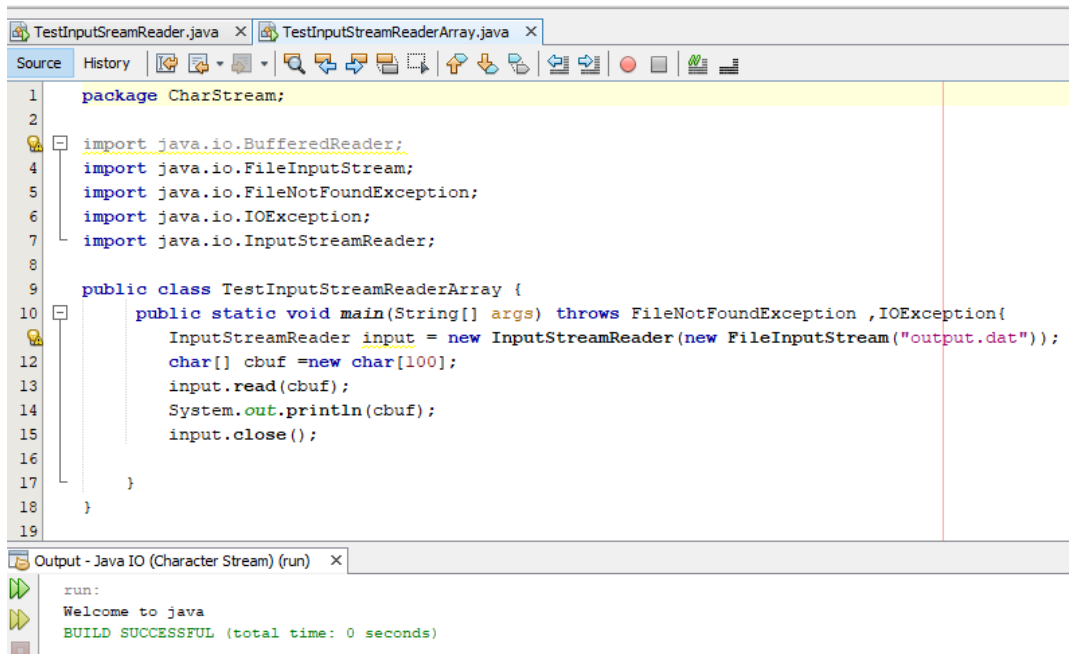
Out/Input Stream



The screenshot shows an IDE with the file `TestOutputStreamWriter.java` open. The code defines a package `CharStream` and imports `java.io` classes: `FileNotFoundException`, `FileOutputStream`, `IOException`, and `OutputStreamWriter`. A public class `TestOutputStreamWriter` contains a `main` method that writes the string "Welcome to java" to a file named `output.dat` and then closes the writer. Below the code editor, the output window shows the result of running the program: "Welcome to javaBUILD SUCCESSFUL (total time: 0 seconds)".

```
1 package CharStream;
2
3
4 import java.io.FileNotFoundException;
5 import java.io.FileOutputStream;
6 import java.io.IOException;
7 import java.io.OutputStreamWriter;
8
9 public class TestOutputStreamWriter {
10
11
12     public static void main(String[] args) throws FileNotFoundException ,IOException{
13         String data = "Welcome to java";
14         OutputStreamWriter output = new OutputStreamWriter(new FileOutputStream("output.dat"));
15         output.write(data);
16         output.close();
17     }
18
19 }
```

run:
Welcome to javaBUILD SUCCESSFUL (total time: 0 seconds)

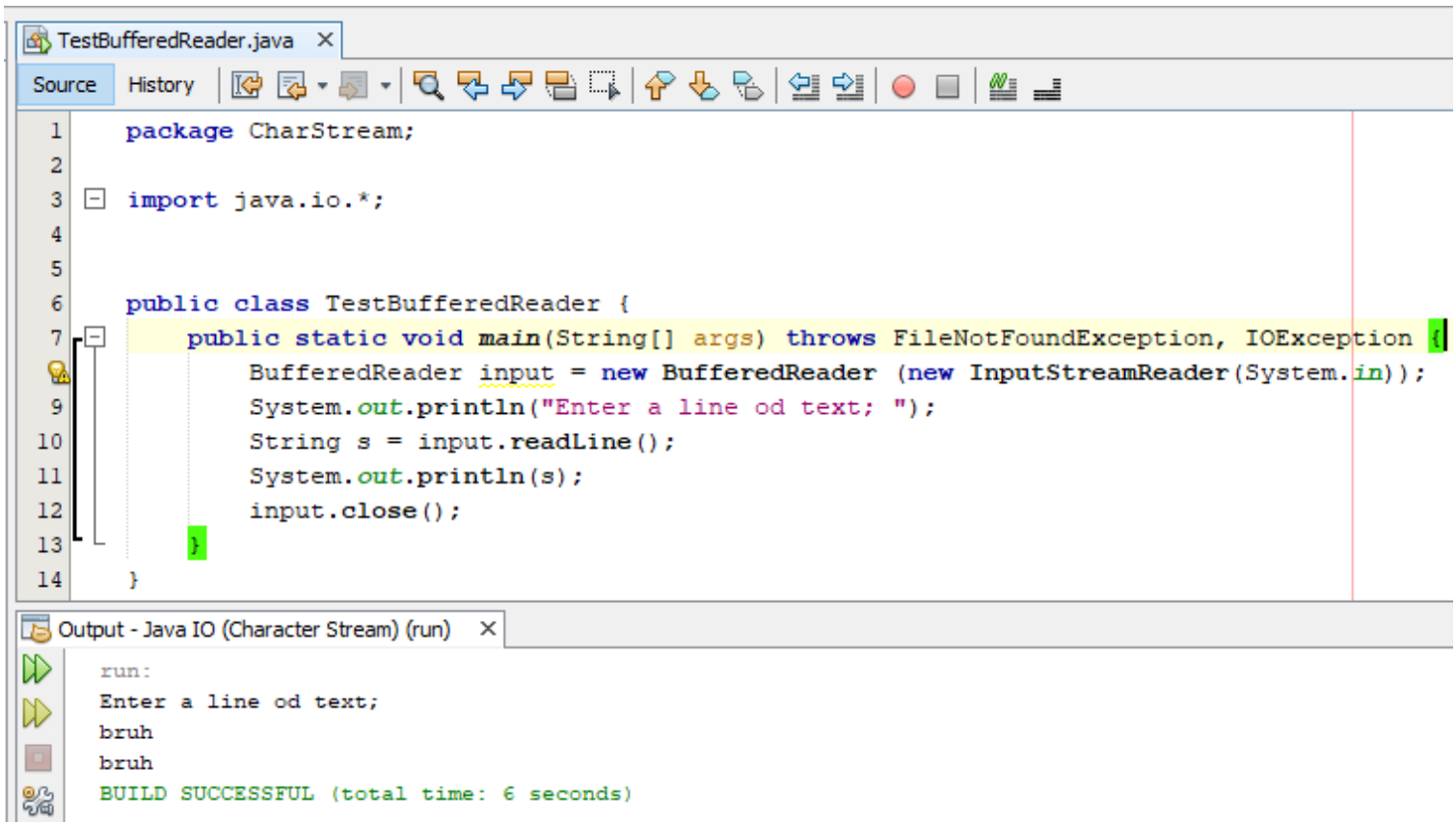


The screenshot shows an IDE with the file `TestInputStreamReaderArray.java` open. The code defines a package `CharStream` and imports `java.io` classes: `BufferedReader`, `FileInputStream`, `FileNotFoundException`, `IOException`, and `InputStreamReader`. A public class `TestInputStreamReaderArray` contains a `main` method that reads the contents of `output.dat` into a character buffer and prints it to the console. Below the code editor, the output window shows the result of running the program: "Welcome to javaBUILD SUCCESSFUL (total time: 0 seconds)".

```
1 package CharStream;
2
3
4 import java.io.BufferedReader;
5 import java.io.FileInputStream;
6 import java.io.FileNotFoundException;
7 import java.io.IOException;
8 import java.io.InputStreamReader;
9
10 public class TestInputStreamReaderArray {
11
12     public static void main(String[] args) throws FileNotFoundException ,IOException{
13         InputStreamReader input = new InputStreamReader(new FileInputStream("output.dat"));
14         char[] cbuf =new char[100];
15         input.read(cbuf);
16         System.out.println(cbuf);
17         input.close();
18     }
19 }
```

run:
Welcome to java
BUILD SUCCESSFUL (total time: 0 seconds)

Buffered read system.in

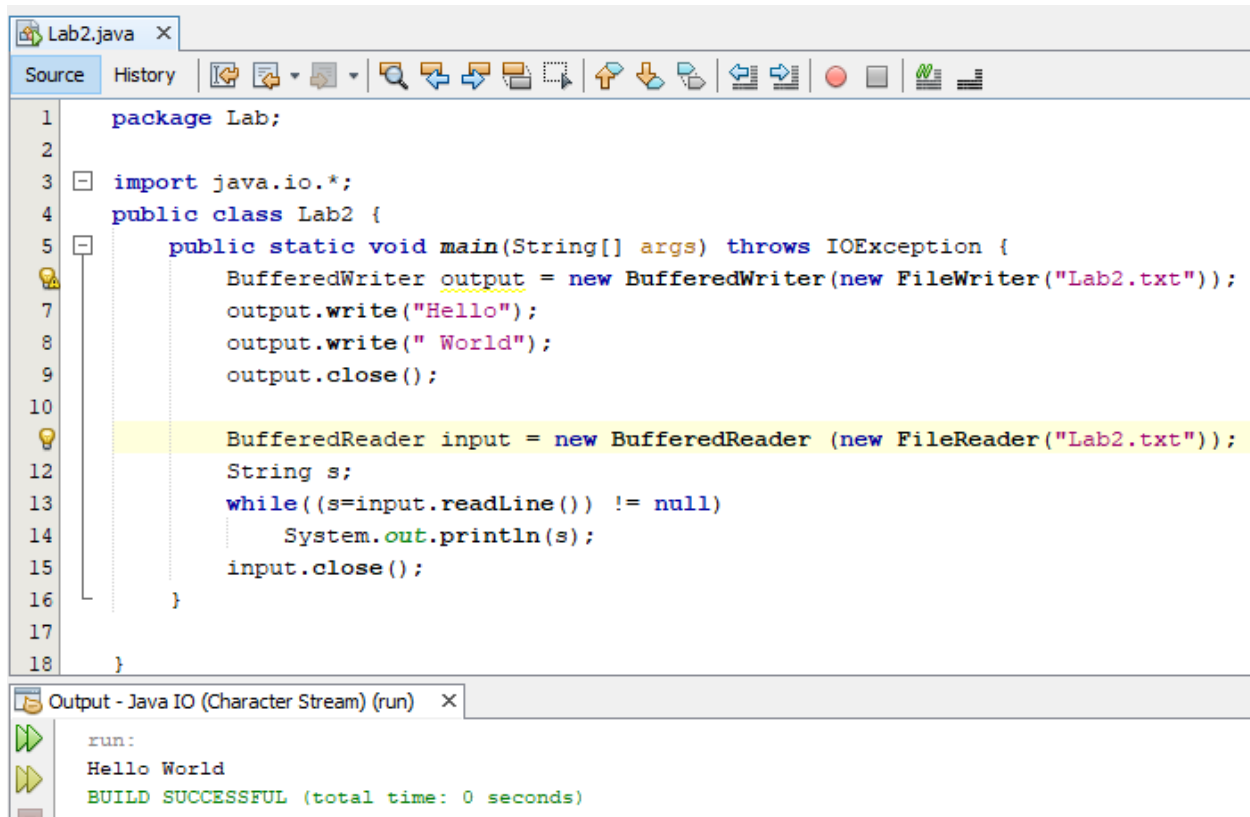


```
1 package CharStream;
2
3 import java.io.*;
4
5
6 public class TestBufferedReader {
7     public static void main(String[] args) throws FileNotFoundException, IOException {
8         BufferedReader input = new BufferedReader (new InputStreamReader(System.in));
9         System.out.println("Enter a line od text; ");
10        String s = input.readLine();
11        System.out.println(s);
12        input.close();
13    }
14 }
```

Output - Java IO (Character Stream) (run) X

run:
Enter a line od text;
bruh
bruh
BUILD SUCCESSFUL (total time: 6 seconds)

BufferedWriter/Reader



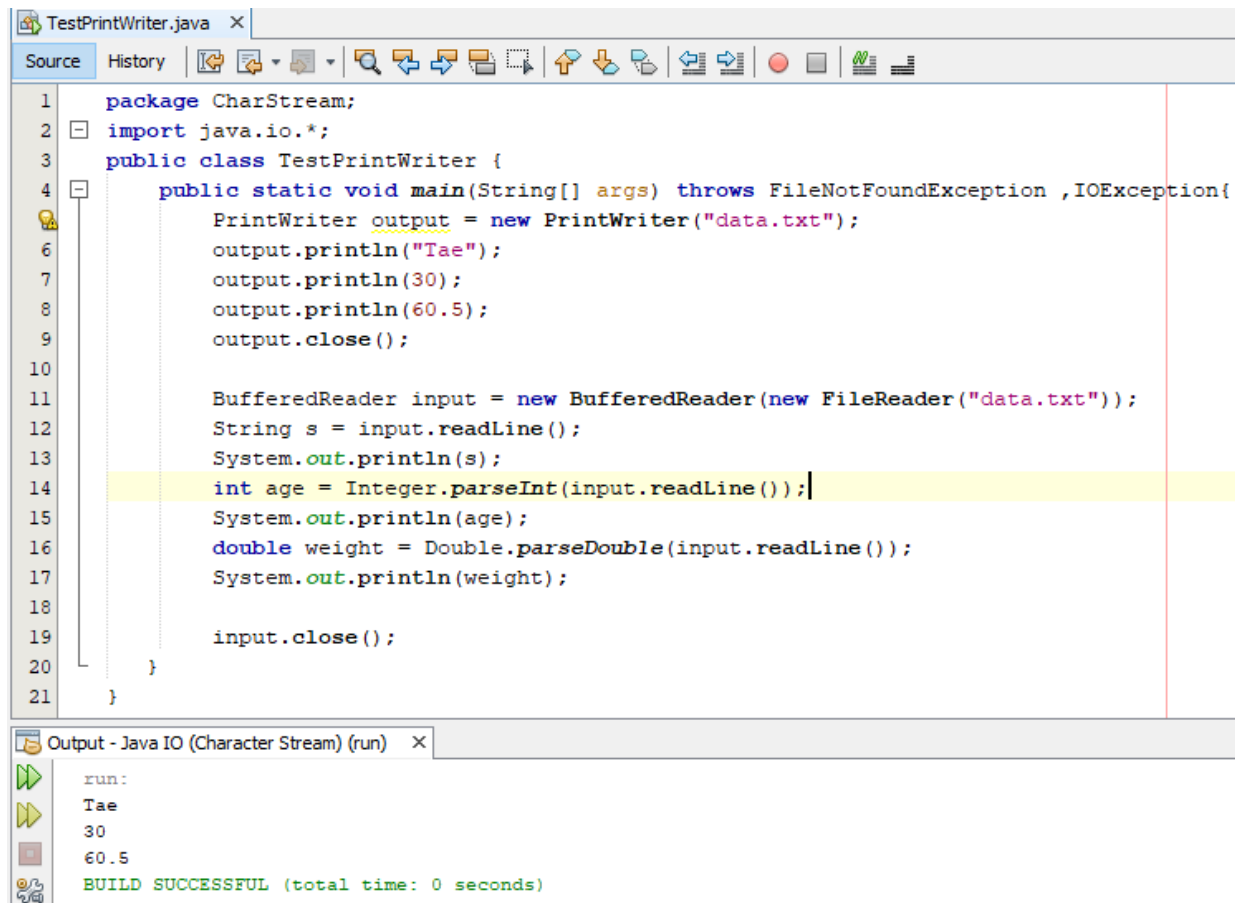
The screenshot shows an IDE window titled "Lab2.java" with a source code editor and an output console. The code defines a package "Lab" and a class "Lab2" with a "main" method. The "main" method uses "BufferedWriter" to write "Hello" and "World" to "Lab2.txt", and "BufferedReader" to read the contents back. The output console shows the program running successfully and printing "Hello World".

```
1 package Lab;
2
3 import java.io.*;
4 public class Lab2 {
5     public static void main(String[] args) throws IOException {
6         BufferedWriter output = new BufferedWriter(new FileWriter("Lab2.txt"));
7         output.write("Hello");
8         output.write(" World");
9         output.close();
10
11         BufferedReader input = new BufferedReader (new FileReader("Lab2.txt"));
12         String s;
13         while ((s=input.readLine()) != null)
14             System.out.println(s);
15         input.close();
16     }
17 }
18 }
```

Output - Java IO (Character Stream) (run) ×

```
run:
Hello World
BUILD SUCCESSFUL (total time: 0 seconds)
```

PrintWriter

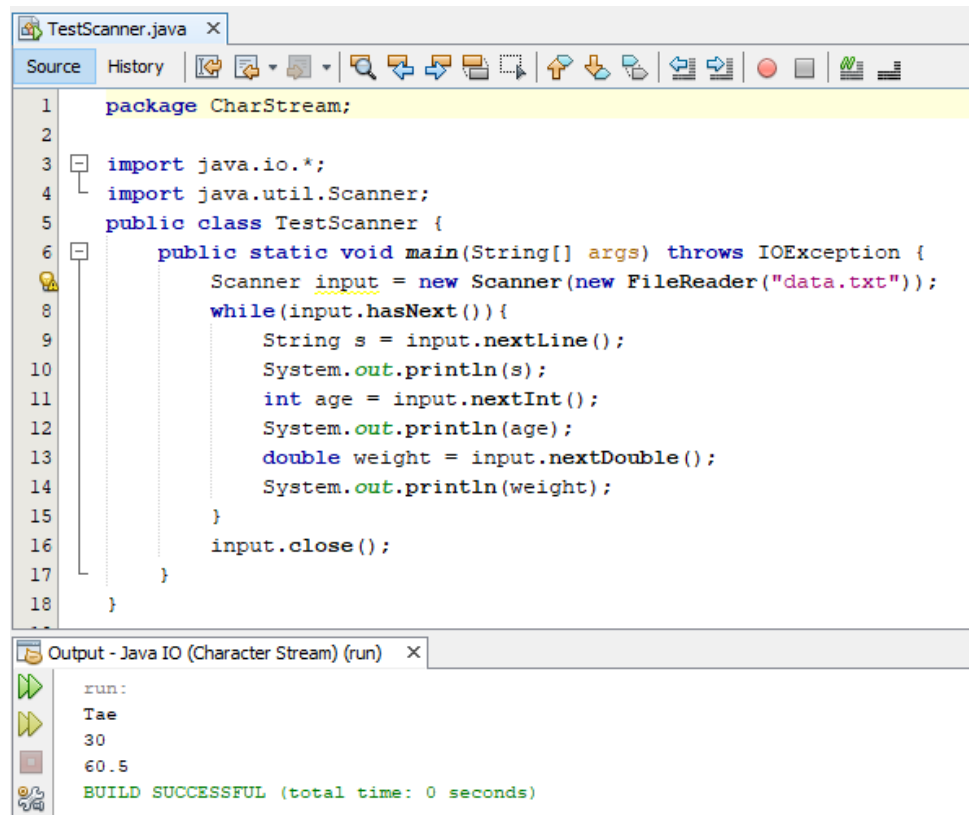


```
1 package CharStream;
2 import java.io.*;
3 public class TestPrintWriter {
4     public static void main(String[] args) throws FileNotFoundException ,IOException{
5         PrintWriter output = new PrintWriter("data.txt");
6         output.println("Tae");
7         output.println(30);
8         output.println(60.5);
9         output.close();
10
11         BufferedReader input = new BufferedReader(new FileReader("data.txt"));
12         String s = input.readLine();
13         System.out.println(s);
14         int age = Integer.parseInt(input.readLine());
15         System.out.println(age);
16         double weight = Double.parseDouble(input.readLine());
17         System.out.println(weight);
18
19         input.close();
20     }
21 }
```

Output - Java IO (Character Stream) (run)

run:
Tae
30
60.5
BUILD SUCCESSFUL (total time: 0 seconds)

Scanner



The screenshot shows an IDE window titled "TestScanner.java". The code defines a package "CharStream" and imports "java.io.*" and "java.util.Scanner". A public class "TestScanner" contains a "main" method that throws "IOException". Inside "main", a "Scanner" object named "input" is created from a "FileReader" reading "data.txt". A "while" loop with "input.hasNext()" condition processes the file content. The loop reads a line of text, an integer age, and a double weight, printing each to "System.out". After the loop, "input.close()" is called. The bottom panel shows the output of the program: "run:", "Tae", "30", "60.5", and a "BUILD SUCCESSFUL" message.

```
1 package CharStream;
2
3 import java.io.*;
4 import java.util.Scanner;
5 public class TestScanner {
6     public static void main(String[] args) throws IOException {
7         Scanner input = new Scanner(new FileReader("data.txt"));
8         while(input.hasNext()) {
9             String s = input.nextLine();
10            System.out.println(s);
11            int age = input.nextInt();
12            System.out.println(age);
13            double weight = input.nextDouble();
14            System.out.println(weight);
15        }
16        input.close();
17    }
18 }
```

Output - Java IO (Character Stream) (run) x

run:
Tae
30
60.5
BUILD SUCCESSFUL (total time: 0 seconds)