Java Assignment4
on Datatype conversion

YANATI RAJANIVAS

CONTACT

rajanivas9573@gmail.com

+919573627556

http://www.linkedin.com/in/yanati-rajanivas-96004232

1-164 Kalavakonda, Chillakur Mandal(%24412), Tirupati Diostrict.

SKILLS

- Python, Java Basics
- Html and Css
- Javascript
- Sql
- Database Fundamentals

EDUCATION

- **Bachelor Of Technology (6.8 CGPA)**
  NBKR Institute of Science and Technology
  2019 - 2023
  Vidyanagar,Nellore
  Computer Science Engineering
- **INTERMEDIATE (9.1 CGPA)**
  NBKR Science And Arts College
  2017 - 2019
  Vidyanagar,Nellore
  MPC
- **S.S.C (8.2 CGPA)**
  Z.P.P High School
  2016 - 2017
  Thikkavaram, Nellore

LANGUAGES

- English
- Telugu

CAREER OBJECTIVE

Highly motivated and enthusiastic to move in software industry as a graduate seeking an opportunity to apply my knowledge, skills, and passion in a dynamic work environment. Eager to contribute to organization by utilizing my technical knowledge and gaining practical experience in work. Committed to continuous learning and growth, with a strong dedication to achieving professional excellence.

PROJECTS AND INTERNSHIPS

Projects

- **Design Of secured Authenticated Key Management Protocol For Cloud computing Environments**

  Front End : HTML , CSS, Javascript **Back End**: JSP

  Data base: MYSQL 5.0 **Server**: Apache Tomcat

Internships

- **Web Development using Html and Css**

  company : Think -Champ pvt.Ltd **Duration:** 1 Month

  Project : Flower Shop Through online
- **Completed internship in machine learning with python**

  company : Verzeo Edutech **Duration:** 1 Month

  Project: Heart Disease prediction using Machine Learning

ACHIEVEMENTS

- ✓ Second place winner in district level drawing competition
- ✓ Won second prize in ball batmenton game under district level

STRENGTHS

- ➢ Research and Analysis
- ➢ Quick learner
- ➢ Positivity and Adaptability
- ➢ Team Player

CERTIFICATIONS

- ❖ Python
- ❖ Html And Css
- ❖ Machine Learning
- ❖ Azure AI Fundamentals
- ❖ Database Administrator Fundamentals

1) Char to Char Datatype Conversion:

In Java, you don't need explicit type casting when converting a **char** to another **char**, as they are both of the same data type. A **char** represents a single 16-bit Unicode character, and it can be assigned to another **char** variable directly.

Result:

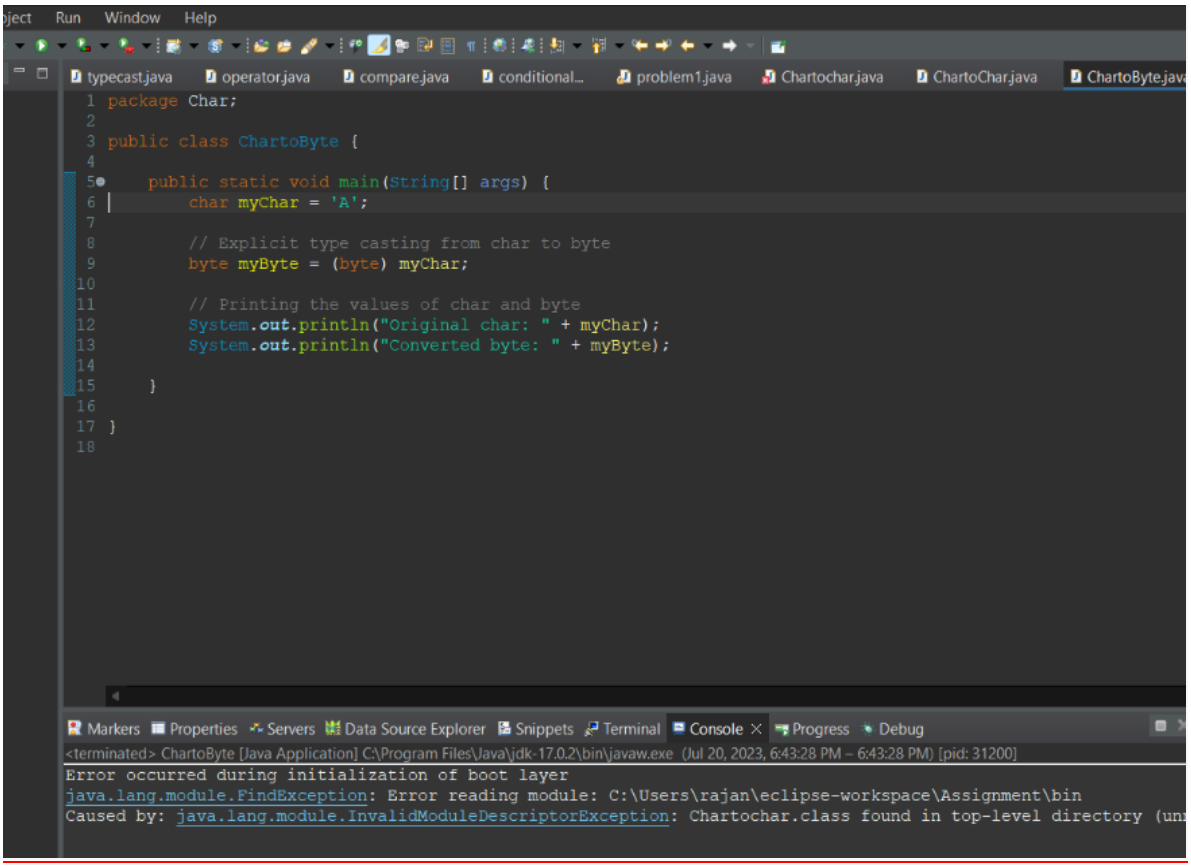
No conversion required for char to char.

2) Char to Byte Datatype Conversion:

In Java, you can perform explicit type casting to convert a **char** to a **byte**. The **char** data type represents a 16-bit Unicode character, whereas the **byte** data type is an 8-bit signed integer.

Since a **char** requires 16 bits to store a character, and a **byte** can only store 8 bits, you need to be cautious when performing this type casting. If the **char** value being casted is outside the valid range of a **byte**, it may result in data loss or unexpected behavior.

Example:



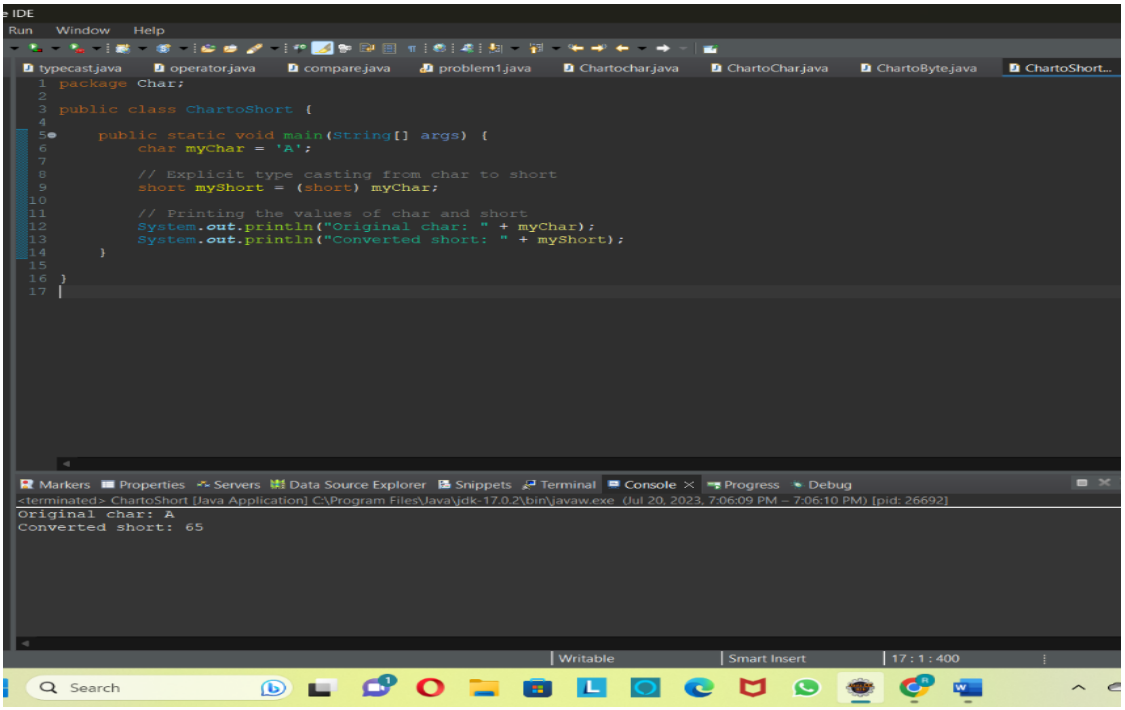
Output:

Error Occurred and it may result in data loss because it is explicit type conversion.

3)Char to Short Data Conversion:

In Java, It needs to perform explicit type casting to convert a **char** to a **short**. The **char** data type represents a 16-bit Unicode character, whereas the **short** data type is a 16-bit signed integer.

Example:



Result:

It is done explicit type conversion both **char** and **short** are 16-bit data types in Java, so the type casting can be done directly without any loss of data or precision.

4) Char to Int Data conversion:

In Java, you can perform an implicit type conversion from **char** to **int** because **char** is a smaller data type than **int**. The **char** data type represents a 16-bit Unicode character, whereas the **int** data type is a 32-bit signed integer.

When converting a **char** to an **int**, the Unicode code point of the character is used as the integer value.

Example:

```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

5) Char to Long Type Conversion:

In Java, you can perform an implicit type conversion from **char** to **long**. The **char** data type represents a 16-bit Unicode character, whereas the **long** data type is a 64-bit signed integer.

where data is converted from a smaller data type (**char**) to a larger data type (**long**) without any data loss.

Example

```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

:

6) Char to Float conversion:

In Java, you can perform an implicit type conversion from **char** to **float**. The **char** data type represents a 16-bit Unicode character, whereas the **float** data type is a 32-bit single-precision floating-point number.

Example:

```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

7) Char to Double conversion:

In Java, you can perform an explicit type conversion from **char** to **double**. The **char** data type represents a 16-bit Unicode character, whereas the **double** data type is a 64-bit double-precision floating-point number.

Example:

```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

8) Char to Boolean conversion:

In Java, you cannot directly cast a **char** to a **boolean** because they are incompatible data types. The **char** data type represents a 16-bit Unicode character, whereas the **boolean** data type can only hold **true** or **false**.

f you want to convert a **char** value to a **boolean**, you need to define a condition or a rule to determine whether the **char** value should be treated as **true** or **false**.

Result:

None

Byte Type Conversions:

1) Byte to Char conversion:

In Java, converting a **byte** to a **char** involves explicit type casting because **byte** is an integral type, and **char** is a character data type.

Example:

```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

2) Byte to byte conversion:

Converting a **byte** to another **byte** doesn't require any explicit type casting, as it's already the same data type.

Result:

No conversion required for char to char.

3) Byte to short conversion:

Converting a **byte** to a **short** involves implicit type conversion, as the **short** data type can safely hold the range of values that can be represented by a **byte**.

Example:

```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

4) Byte to int conversion:

Converting a **byte** to an **int** involves implicit type conversion, as the **int** data type can safely hold the range of values that can be represented by a **byte**.

Example:

```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

5) Byte to Long conversion:

Converting a **byte** to a **long** involves implicit type conversion, as the **long** data type can safely hold the range of values that can be represented by a **byte**.

Example:


```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

6) Byte to Float Conversion:

Converting a **byte** to a **float** involves implicit type conversion, as the **float** data type can safely hold the range of values that can be represented by a **byte**.

Example:

```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

7) Byte to Double conversion:

Converting a **Byte** wrapper object or a **byte** primitive to a **double** involves implicit type conversion, as the **double** data type can safely hold the range of values that can be represented by a **Byte** or **byte**.

Example:

.

```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

8)Byte to Boolean conversion:

In Java, there is no direct conversion between **byte** and **boolean** data types. Converting a **byte** to a **boolean** cannot be done implicitly or explicitly through type casting, as they are not compatible types.

Result: None

1)Short to Char conversion:

In Java, converting a **short** to a **char** involves explicit type casting because **short** is an integral data type, and **char** is a character data type.

Example:

```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

2) Short to Byte conversion:

In Java, converting a **`short`** to a **`byte`** involves explicit type casting because **`short`** is a 16-bit integral data type, and **`byte`** is an 8-bit integral data type. Therefore, a narrowing conversion is required, and you need to explicitly cast the **`short`** value to a **`byte`**.

Example:

```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

3) Short to Short conversion:

Converting a **short** to another **short** doesn't require any explicit type casting, as it's already the same data type. It is essentially an identity conversion, meaning the value remains the same.

Result: None

4) Short to int conversion:

Converting a **short** to an **int** involves implicit type conversion, as the **int** data type can safely hold the range of values that can be represented by a **short**.

Example:

```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

5) Short to Long conversion:

In Java, converting a **short** to an **int** or a **long** involves implicit type conversion, as both **int** and **long** data types can safely hold the range of values that can be represented by a **short**.

Example:


```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

6) Short to float conversion:

In Java, converting a **short** to a **float** involves implicit type conversion, as the **float** data type can safely hold the range of values that can be represented by a **short**.

Example:

```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

7) Short to Double conversion:

In Java, converting a **short** to a **double** involves implicit type conversion, as the **double** data type can safely hold the range of values that can be represented by a **short**.

Example:

```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe  (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

8) Short to boolean conversion:

n Java, there is no direct conversion between **short** and **boolean** data types. Converting a **short** to a **boolean** cannot be done implicitly or explicitly through type casting, as they are not compatible types.

Result: None

1) Int to Char conversion in java:

In Java, converting an **int** to a **char** involves explicit type casting because **int** is an integral data type, and **char** is a character data type.

Example:

```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

2) Int to Byte conversion:

In Java, converting an **int** to a **byte** involves explicit type casting because **int** is a 32-bit integral data type, and **byte** is an 8-bit integral data type. Therefore, a narrowing conversion is required, and you need to explicitly cast the **int** value to a **byte**.

Example:

```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

3) Int to Short conversion:

In Java, converting an **int** to a **short** involves explicit type casting because **int** is a 32-bit integral data type, and **short** is a 16-bit integral data type. Therefore, a narrowing conversion is required, and you need to explicitly cast the **int** value to a **short**.

Example:


```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

4)Int to Int conversion:

It seems there might be a typo in your question. If you meant "int to int" conversion in Java, then that would be a simple identity conversion where no explicit type casting is required, as **int** is already an **int**.

Result: Not required

5) Int to Long conversion:

In Java, converting an **int** to a **long** involves implicit type conversion, as the **long** data type can safely hold the range of values that can be represented by an **int**.

Example:

```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

6) Int to Float conversion:

In Java, converting an **int** to a **float** involves implicit type conversion, as the **float** data type can safely hold the range of values that can be represented by an **int**.

Example:

```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

7)Int to Double conversion:

In Java, converting an **int** to a **double** involves implicit type conversion, as the **double** data type can safely hold the range of values that can be represented by an **int**.

Example:


```java
package Char;

public class ChartoInt {

    public static void main(String[] args) {
        char myChar = 'A';

        // Implicit type conversion from char to int

        int myInt = myChar;

        // Printing the values of char and int
        System.out.println("Original char: " + myChar);
        System.out.println("Converted int: " + myInt);

    }

}
```

```
<terminated> ChartoChar [Java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (Jul 20, 2023, 7:23:59 PM – 7:23:59 PM) [pid: 20412]
First character: A
Second character: A
```

8) Int to Boolean conversion:

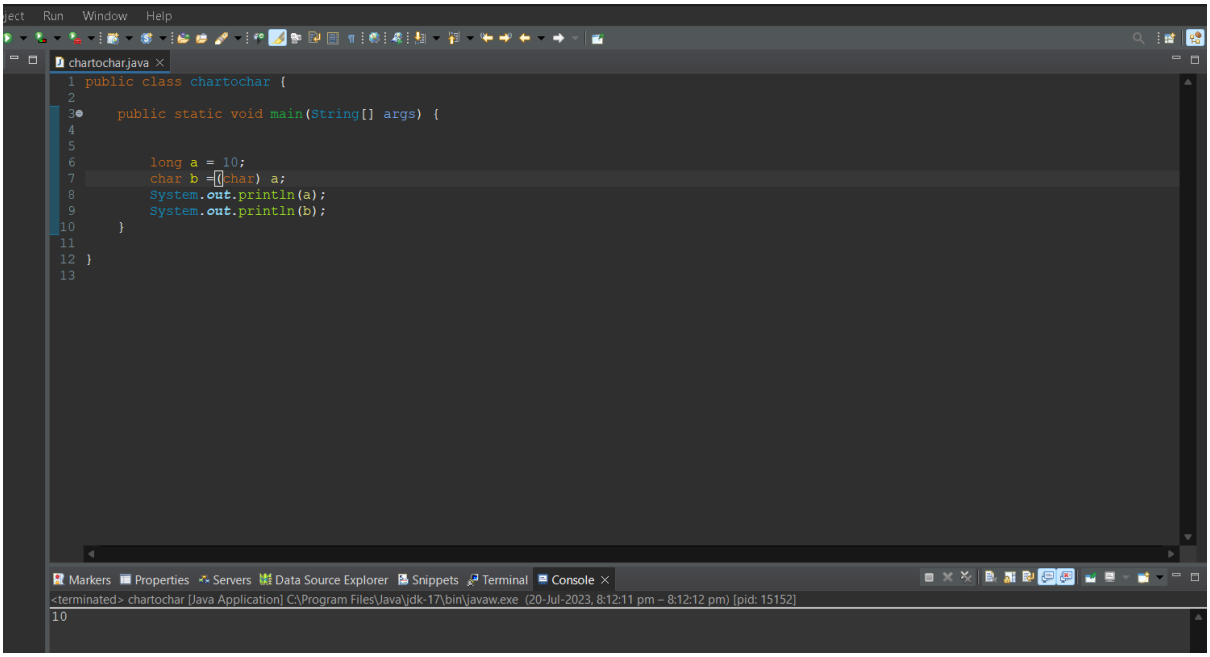
In Java, there is no direct conversion between **int** and **boolean** data types. Converting an **int** to a **boolean** cannot be done implicitly or explicitly through type casting, as they are not compatible types.

Result: No conversion

1) Long to char explicit :

In Java, you can convert a **long** data type to a **char** data type using type casting or by converting the **long** to a **String** first and then getting the **char** from the **String**. Let's see both methods:

Example:



2) Long to byte explicit:

In Java, you can convert a **long** data type to a **byte** data type using type casting. However, be aware that a **long** value may not always fit into a **byte**, as **long** can store much larger values than **byte**. If the **long** value is outside the valid range of a **byte**, data loss may occur. To avoid potential data loss, you should

perform proper validation before the conversion.

```java
public class chartochar {

    public static void main(String[] args) {


        long a = 10;
        byte b = (byte)a;
        System.out.println(a);
        System.out.println(b);
    }

}
```

```
<terminated> chartochar [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe  (20-Jul-2023, 8:13:36 pm – 8:13:36 pm) [pid: 21724]
10
10
```

3) Long to short explicit:

In Java, you can convert a **long** data type to a **short** data type using type casting. However, similar to the previous example, you need to be cautious about potential data loss since a **long** can store larger values than a **short**. You should ensure that the **long** value is within the valid range of a **short** before performing the conversion.

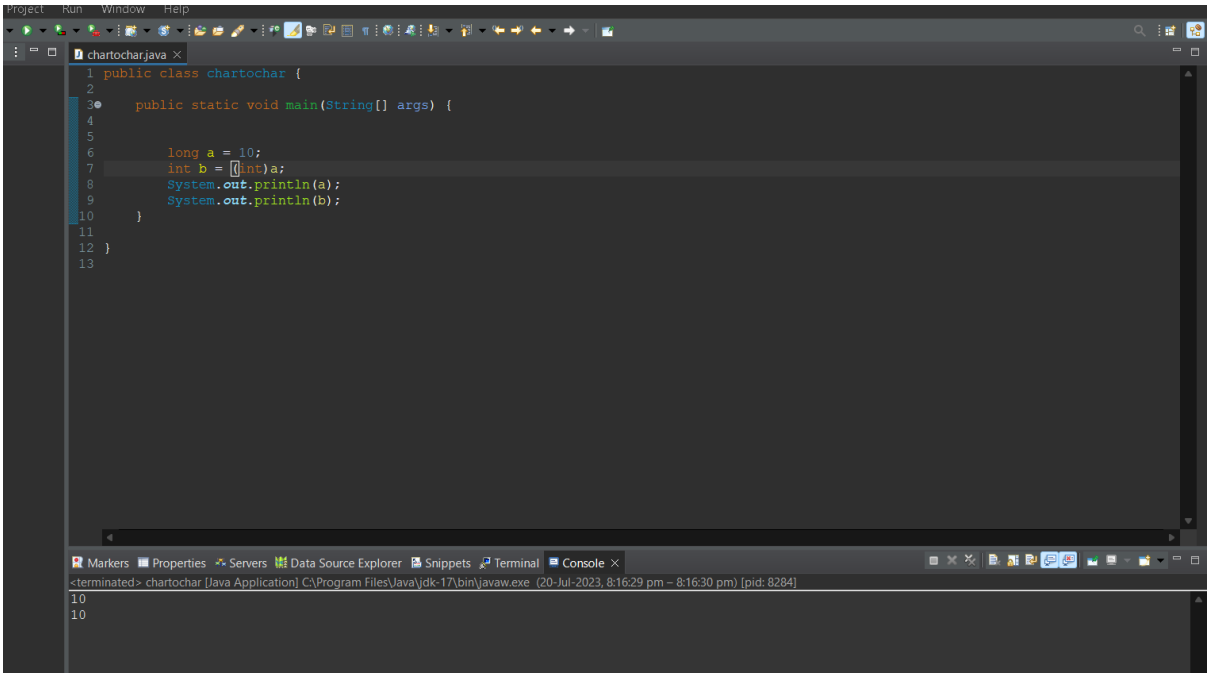
```java
public class chartochar {

    public static void main(String[] args) {


        long a = 10;
        short b =(short) a;
        System.out.println(a);
        System.out.println(b);
    }

}
```

```
<terminated> chartochar [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe  (20-Jul-2023, 8:15:01 pm – 8:15:01 pm) [pid: 20096]
10
10
```

4) Long to int explicit:

In Java, you can convert a **long** data type to an **int** data type using type casting. However, similar to previous examples, you should be careful about potential data loss since a **long** can store larger values than an **int**. You need to ensure that the **long** value is within the valid range of an **int** before performing the conversion.

Example:



5) Long to float implicit :

In Java, you can convert a **long** data type to a **float** data type using type casting. However, you should be aware that type casting from **long** to **float** may result in a loss of precision since **float** is a single-precision 32-bit floating-point data type, whereas **long** is a 64-bit integer data type.

Example:

```java
public class chartochar {

    public static void main(String[] args) {


        long a = 10;
        float b = a;
        System.out.println(a);
        System.out.println(b);
    }

}
```

```
<terminated> chartochar [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe  (20-Jul-2023, 8:17:42 pm – 8:17:42 pm) [pid: 19520]
10
10.0
```

6) Long to double implicit:

In Java, you can convert a **long** data type to a **double** data type using type casting. Converting **long** to **double** does not result in any loss of precision, as **double** is a double-precision 64-bit floating-point data type that can accurately represent a wider range of values, including integers and fractions.

Example:

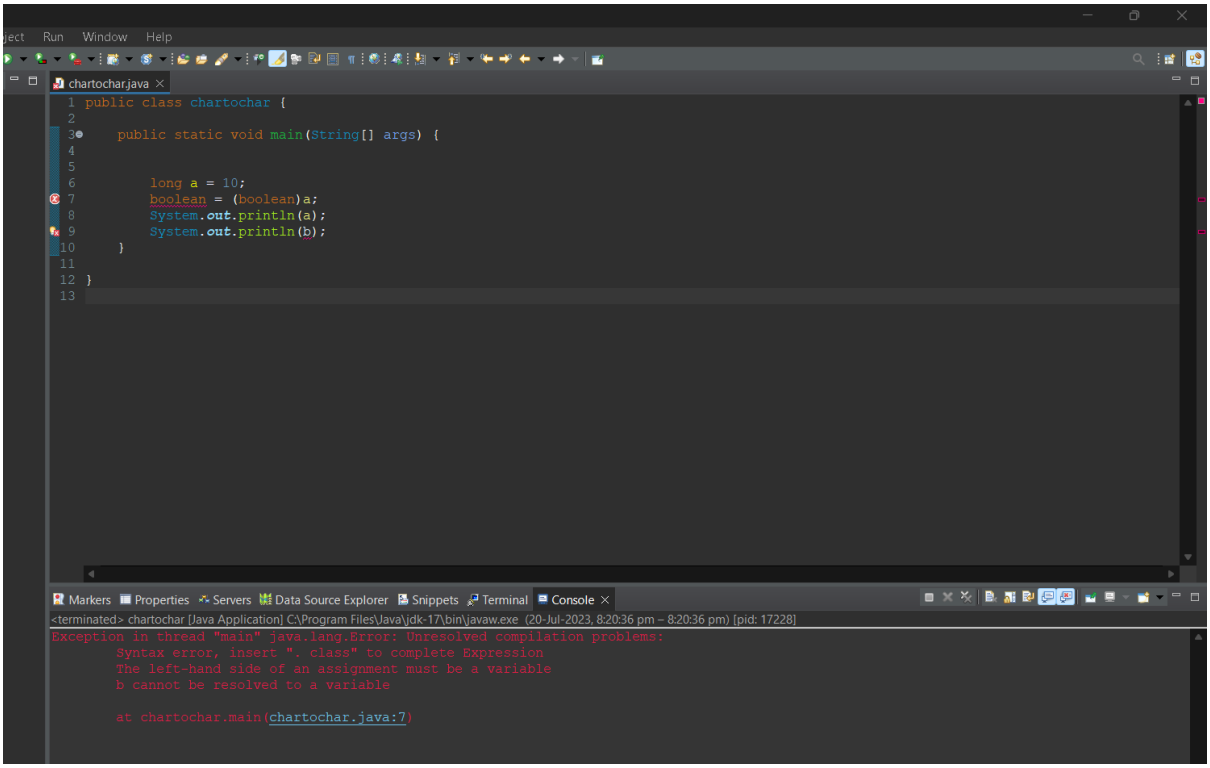
```java
public class chartochar {

    public static void main(String[] args) {


        long a = 10;
        double b = a;
        System.out.println(a);
        System.out.println(b);
    }

}
```

```
<terminated> chartochar [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe  (20-Jul-2023, 8:18:43 pm – 8:18:43 pm) [pid: 7184]
10
10.0
```

7) Long to Boolean:

In Java, you cannot directly convert a **long** data type to a **boolean** data type using type casting, as there is no direct numeric representation of **true** or **false** in a **long**.

If you want to convert a **long** to a **boolean**, you'll need to define a condition based on your specific use case. For example, you can set a rule such that any non-zero value of **long** will be considered **true**, and **false** otherwise.



1)Float to char explicit:

In Java, you cannot directly convert a **float** data type to a **char** data type using type casting, as there is no direct numeric representation of characters in a **float**.

To convert a **float** to a **char**, you can convert the **float** to an integer or a **String** first and then extract the character representation from the integer or the first character of the **String**.

Example:

```java
public class chartochar {

    public static void main(String[] args) {


        float a = 10;
        char  b = (char)a;
        System.out.println(a);
        System.out.println(b);
    }

}
```

```
<terminated> chartochar [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe  (20-Jul-2023, 8:22:53 pm – 8:22:54 pm) [pid: 16584]
10.0
```

2) Float to byte explicit:

In Java, you can convert a **float** data type to a **byte** data type using type casting. However, be aware that converting a **float** to a **byte** may result in a loss of precision, as **byte** is an 8-bit signed integer data type, and **float** is a 32-bit floating-point data type.

Example:

```java
public class chartochar {

    public static void main(String[] args) {


        float a = 10;
        byte  b = (byte)a;
        System.out.println(a);
        System.out.println(b);
    }

}
```

```
<terminated> chartochar [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe  (20-Jul-2023, 8:58:50 pm – 8:58:51 pm) [pid: 15432]
10.0
10
```

3) Float to short explicit:

In Java, you can convert a **float** data type to a **short** data type using type casting. However, similar to previous examples, be aware that converting a **float** to a **short** may result in a loss of precision, as **short** is a 16-bit signed integer data type, and **float** is a 32-bit floating-point data type.

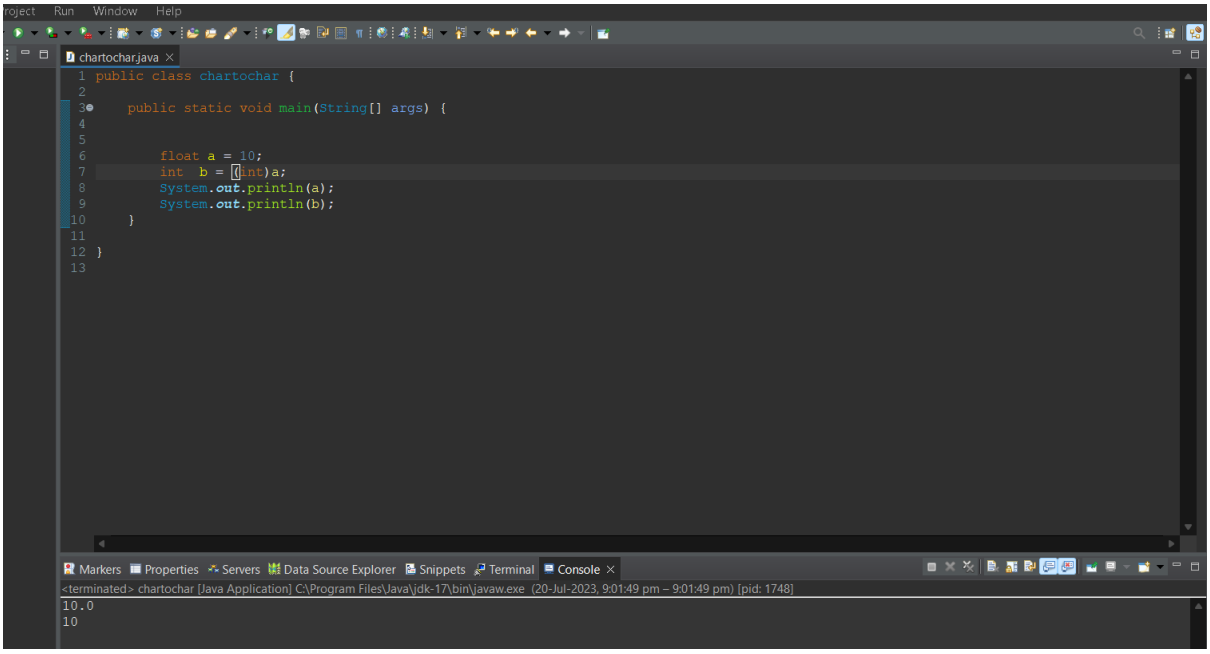
Example:

```java
public class chartochar {

    public static void main(String[] args) {


        float a = 10;
        short b =(short) a;
        System.out.println(a);
        System.out.println(b);
    }

}
```

```
10.0
10
```

4) Float to int explicit :

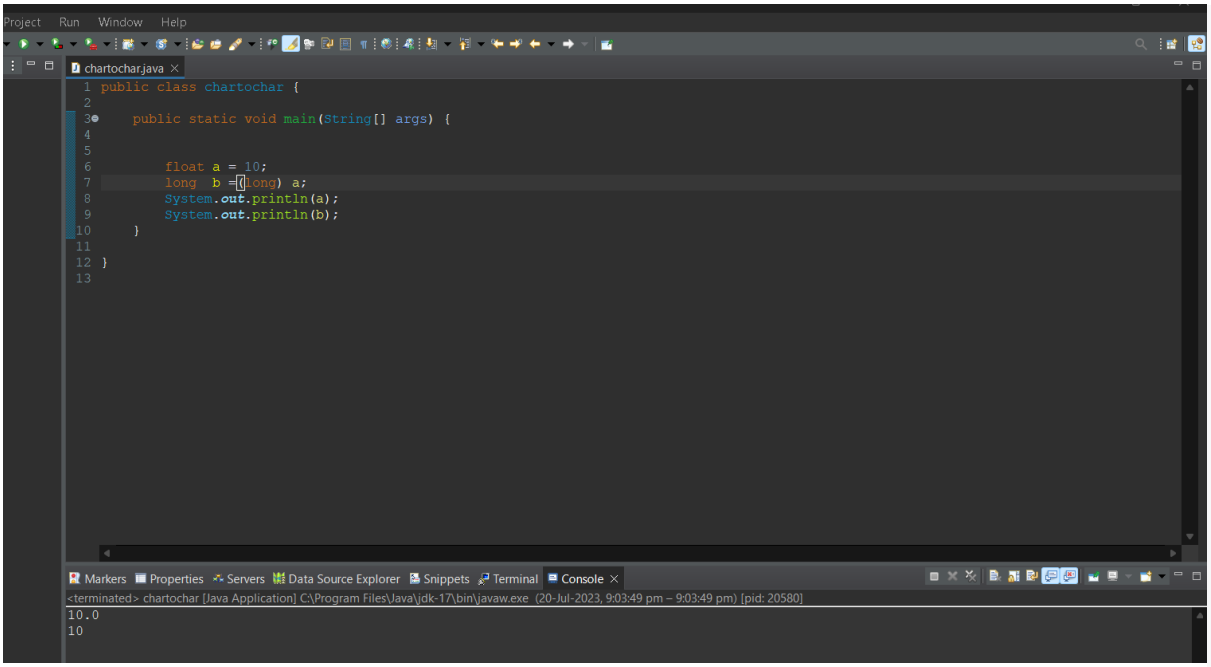
In Java, you can convert a **float** data type to an **int** data type using type casting. However, be aware that converting a **float** to an **int** may result in a loss of precision, as **int** is a 32-bit signed integer data type, and **float** is a 32-bit floating-point data type.



5) Float to long explicit:

In Java, you can convert a **float** data type to a **long** data type using type casting. However, be aware that converting a **float** to a **long** may result in a loss of precision, as **long** is a 64-bit integer data type, and **float** is

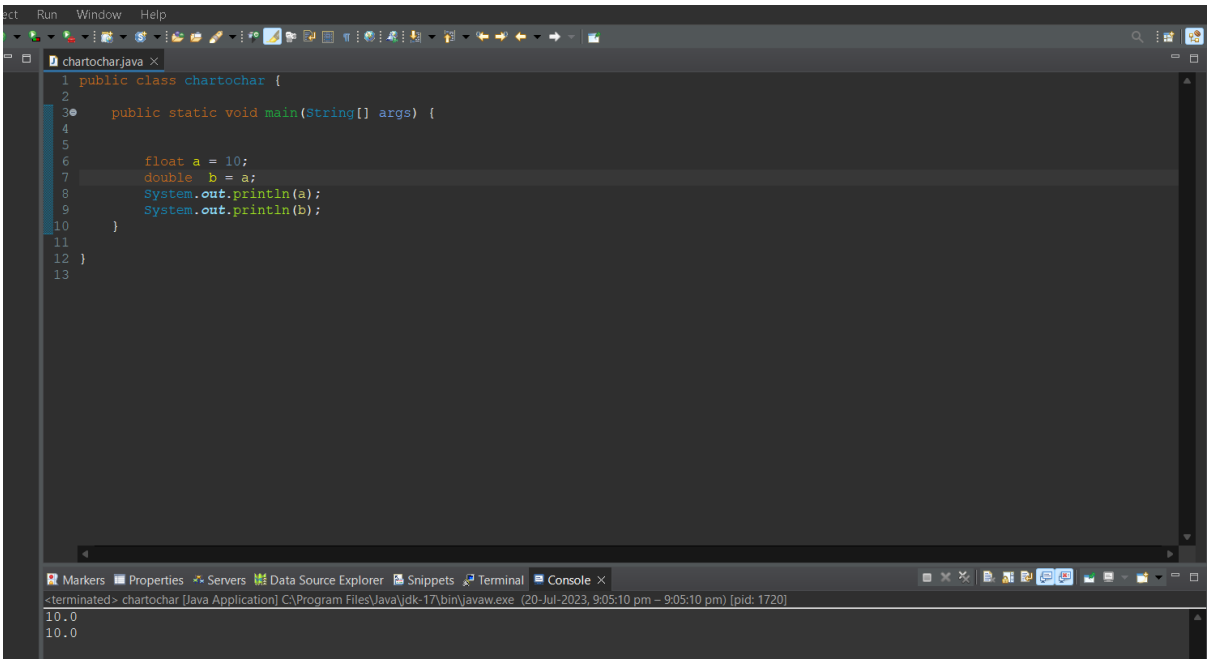
Example:



a 32-bit floating-point data type.

6) Float to double implicit:

In Java, you can convert a **float** data type to a **double** data type without any explicit type casting. This is because **double** is a higher precision floating-point data type than **float**, and Java allows implicit widening conversions from **float** to **double**.



7) Float to Boolean explicit:

In Java, there is no direct conversion from a **float** data type to a **boolean** data type. **boolean** can only represent two values: **true** or **false**. A **float** is a numeric data type and cannot be directly converted to a boolean value.

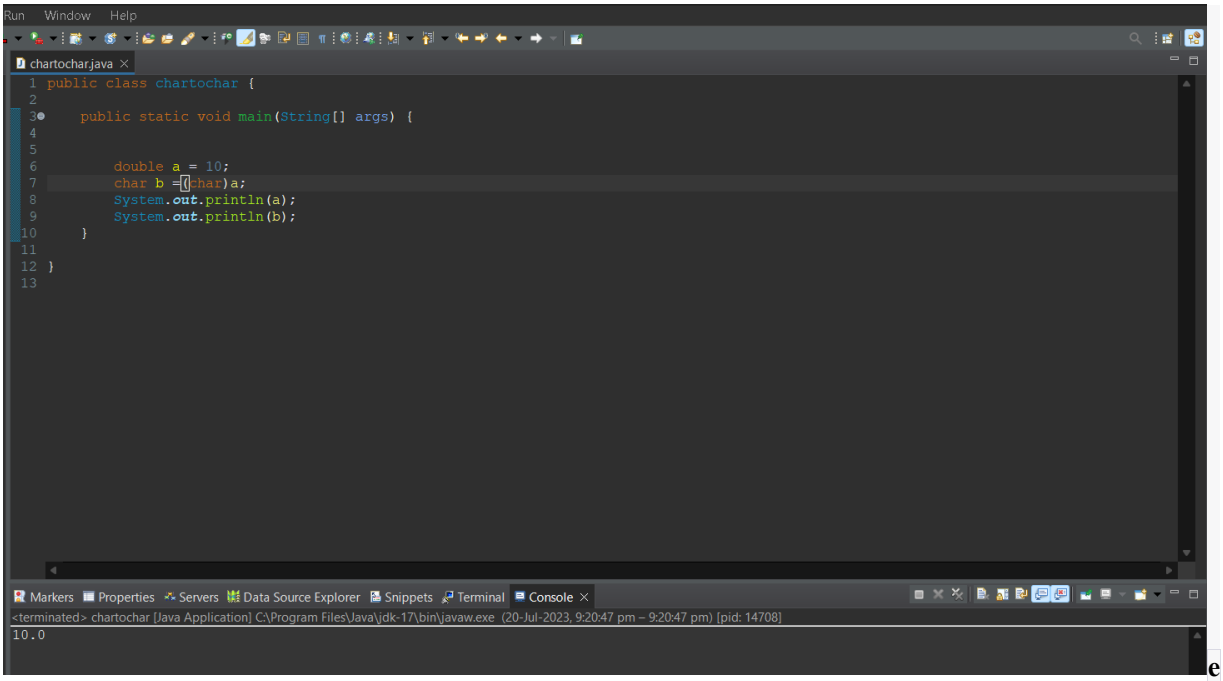
```
public class chartochar {

    public static void main(String[] args) {


        float a = 10;
        boolean  b = (boolean) a;
        System.out.println(a);
        System.out.println(b);
    }

}
```

```
<terminated> chartochar [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe  (20-Jul-2023, 9:06:53 pm – 9:06:54 pm) [pid: 9772]
Exception in thread "main" java.lang.Error: Unresolved compilation problem:
	Cannot cast from float to boolean

	at chartochar.main(chartochar.java:7)
```

1)Double to char explicit:

In Java, you can convert a **double** data type to a **char** data type by first converting the **double** to an integer and then converting the integer to a **char**. This approach assumes that the **doubl** value represents a valid Unicode value for the **char**.

Example:



e

2) Double to byte explicit:

In Java, you can convert a **double** data type to a **byte** data type using type casting. However, be aware that converting a **double** to a **byte** may result in a loss of precision, as **byte** is an 8-bit signed integer data type, and **double** is a 64-bit floating-point data type.

Example:

```java
public class chartochar {

    public static void main(String[] args) {


        double a = 10;
        byte b =(byte) a;
        System.out.println(a);
        System.out.println(b);
    }

}
```

```
<terminated> chartochar [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (20-Jul-2023, 9:22:46 pm – 9:22:46 pm) [pid: 10376]
10.0
10
```

3) Double to short explicit :

In Java, you can convert a **double** data type to a **short** data type using type casting. However, be aware that converting a **double** to a **short** may result in a loss of precision, as **short** is a 16-bit signed integer data type, and **double** is a 64-bit floating-point data type.

Example:

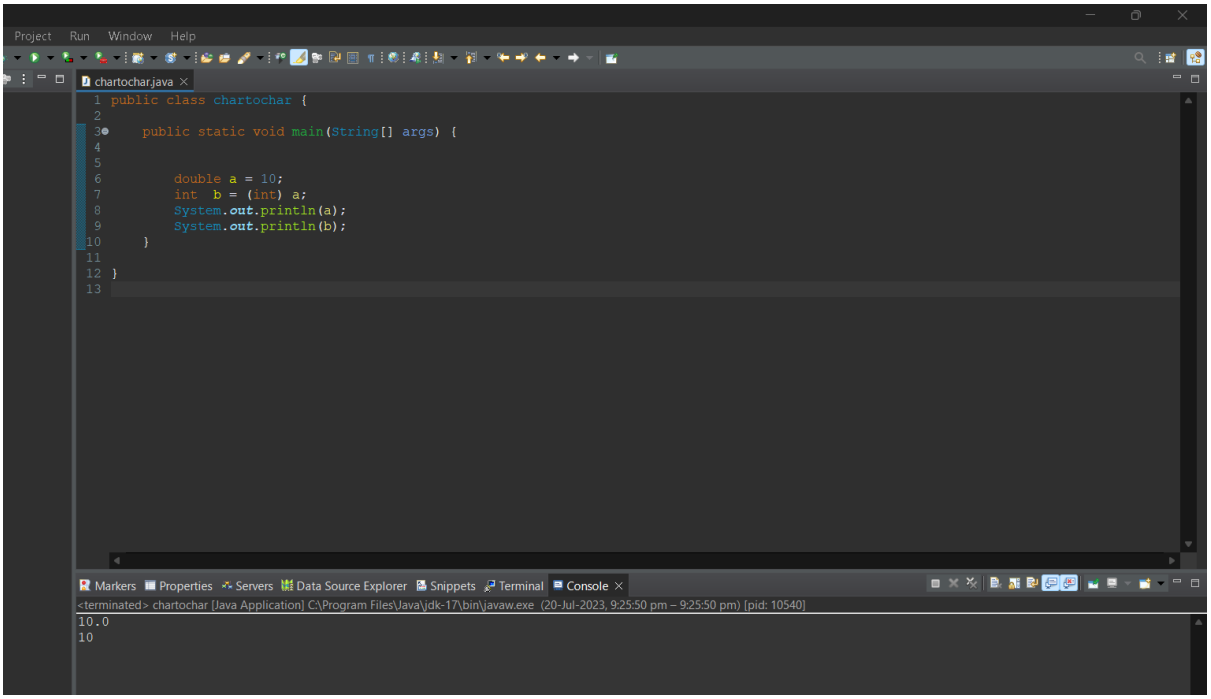
```java
public class chartochar {

    public static void main(String[] args) {


        double a = 10;
        short b = (short)a;
        System.out.println(a);
        System.out.println(b);
    }

}
```

```
<terminated> chartochar [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (20-Jul-2023, 9:24:35 pm – 9:24:35 pm) [pid: 12184]
10.0
10
```

4) Double to int explicit:

In Java, you can convert a **double** data type to an **int** data type using type casting. However, be aware that converting a **double** to an **int** may result in a loss of precision, as **int** is a 32-bit signed integer data type, and **double** is a 64-bit floating-point data type.

Example:



5) Double to long explicit:

In Java, you can convert a **double** data type to a **long** data type using type casting. However, be aware that converting a **double** to a **long** may result in a loss of precision, as **long** is a 64-bit integer data type, and **double** is a 64-bit floating-point data type.

Example:

```java
public class chartochar {

    public static void main(String[] args) {


        double a = 10;
        long b = (long) a;
        System.out.println(a);
        System.out.println(b);
    }

}
```

```
<terminated> chartochar [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe  (20-Jul-2023, 9:27:28 pm – 9:27:28 pm) [pid: 20788]
10.0
10
```

6) Double to float explicit :

In Java, you can convert a **double** data type to a **float** data type using type casting. This conversion is called narrowing conversion, and you may lose precision when converting from **double** to **float**, as **double** is a 64-bit floating-point data type and **float** is a 32-bit floating-point data type.

Example:

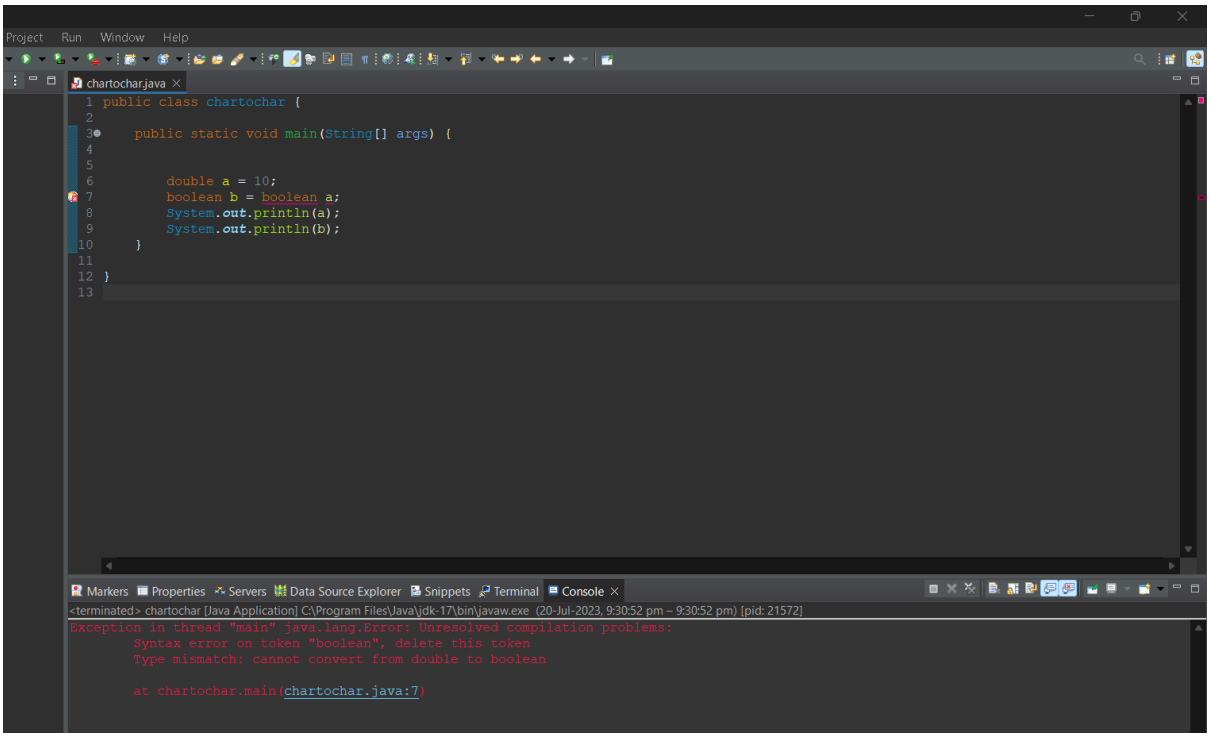
```java
public class chartochar {

    public static void main(String[] args) {


        double a = 10;
        float b = (float) a;
        System.out.println(a);
        System.out.println(b);
    }

}
```

```
<terminated> chartochar [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe  (20-Jul-2023, 9:28:58 pm – 9:28:58 pm) [pid: 13560]
10.0
10.0
```

7) Double to Boolean explicit:

In Java, you cannot directly convert a **double** data type to a **boolean** data type using type casting, as there is no direct numeric representation of **true** or **false** in a **double**.

Example:



1) Boolean to char explicit, 2) **1) Boolean to short explicit**, **3) Boolean to int explicit**,

4) Boolean to long explicit, **5) Boolean to float explicit**, **6) Boolean to double explicit**,

7) Boolean to byte explicit,

In Java, you cannot directly convert a **boolean** data type to a **any** data type using type casting, as there is no direct numeric representation of **true** or **false** in a any datatype.

Table for all datatype conversions

	byte	short	char	int	long	float	double
byte		expl	expl	expl	expl	expl	expl
short	impl		expl	expl	expl	expl	expl
char	expl	expl		expl	expl	expl	expl
int	impl	impl	impl		expl	expl	expl
long	impl	impl	impl	impl		expl	expl
float	impl	impl	impl	impl	impl		expl
double	impl	impl	impl	impl	impl	impl	

Datatype conversion graph