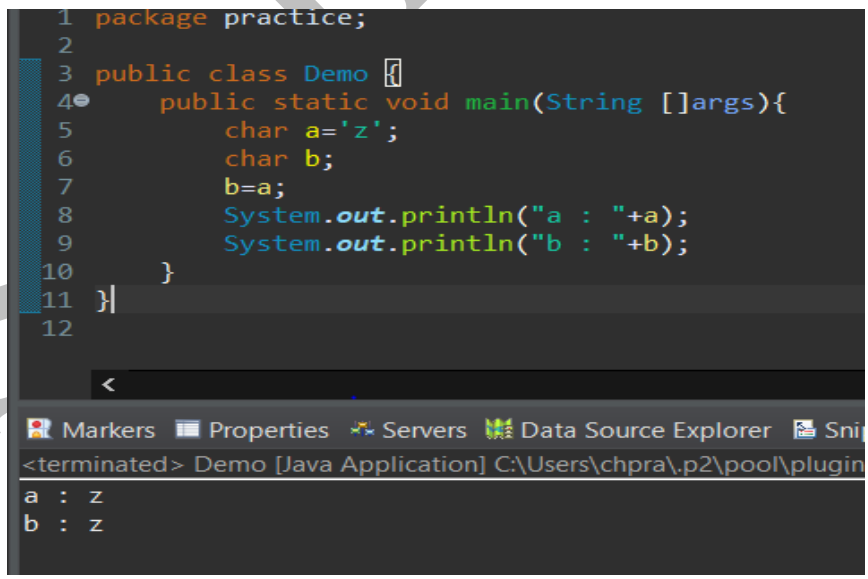
PRIMITIVE DATATYPES TYPECASTING

Conversion of data from char-to-char datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                char a='z';
                char b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:



Conclusion:

char-to-char conversion is not required.

Conversion of data from char-to-byte datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                char a='z';
                byte b;
                b=(byte)a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
package practice;

public class Demo {
    public static void main(String []args){
        char a='z';
        byte b;
        b=(byte)a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer Sn

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plugi
a : z
b : 122
```

Conclusion:

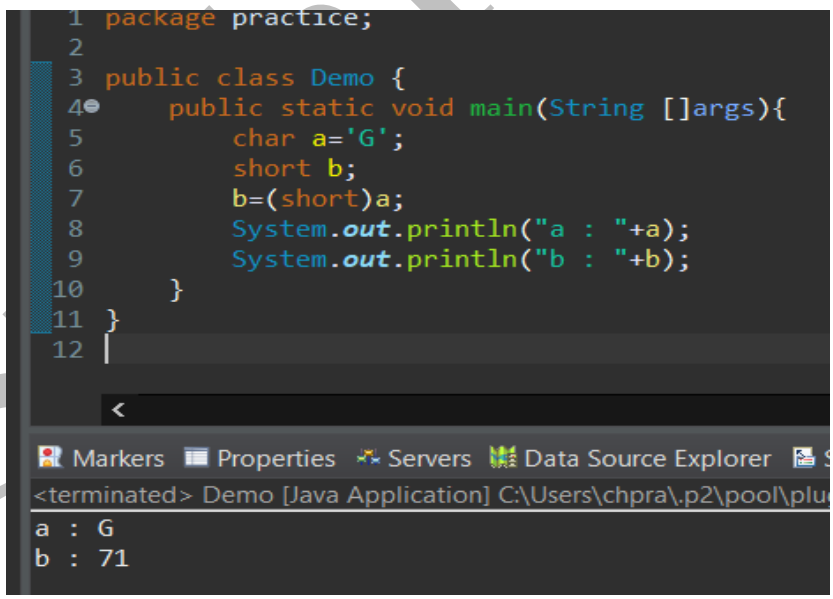
char-to-byte conversion is done through explicit type casting.

Conversion of data from char-to-short datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                char a='G';
                short b;
                b=(short)a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:



Conclusion:

char-to-short conversion is done through explicit type casting.

Conversion of data from char-to-int datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                char a='a';
                int b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
1  package practice;
2
3  public class Demo {
4      public static void main(String []args){
5          char a='a';
6          int b;
7          b=a;
8          System.out.println("a : "+a);
9          System.out.println("b : "+b);
10     }
11 }
12
```

Markers Properties Servers Data Source Explorer S

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plug
a : a
b : 97
```

Conclusion:

char-to-int conversion is done through implicit type casting.

Conversion of data from char-to-long datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                char a='b';
                long b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
package practice;

public class Demo {
    public static void main(String []args){
        char a='b';
        long b;
        b=a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer S

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plug
a : b
b : 98
```

Conclusion:

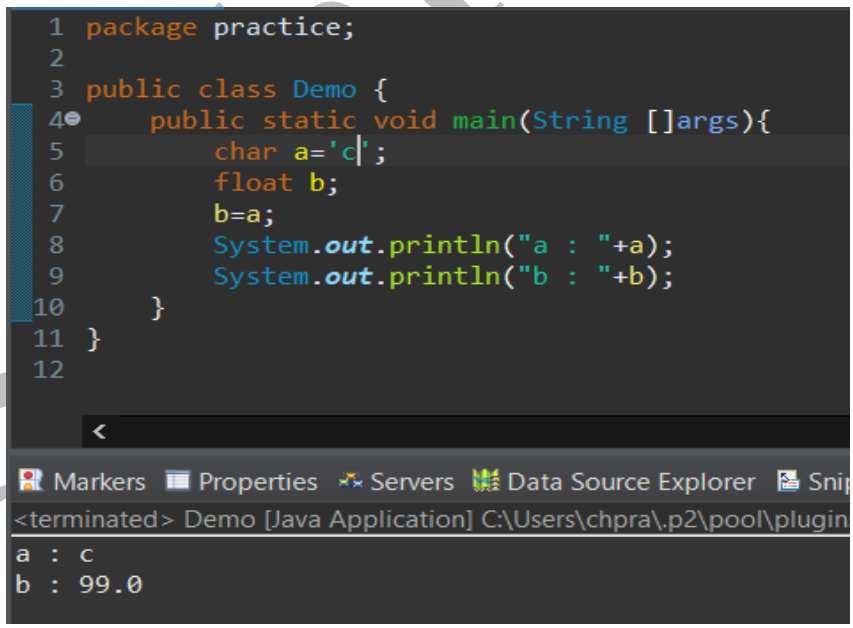
char-to-long conversion is done through implicit type casting.

Conversion of data from char-to-float datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                char a='c';
                float b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:



Conclusion:

char-to-float conversion is done through implicit type casting.

Conversion of data from char-to-double datatype:

Program:

```
package practice;
public class Demo {
	public static void main(String []args){
		char a='d';
		double b;
		b=a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

```
1  package practice;
2
3  public class Demo {
4      public static void main(String []args){
5          char a='d';
6          double b;
7          b=a;
8          System.out.println("a : "+a);
9          System.out.println("b : "+b);
10     }
11 }
12
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plu
a : d
b : 100.0
```

Conclusion:

char-to-double conversion is done through implicit type casting.

Conversion of data from char-to-boolean datatype:

Program://Implicit type casting

```
package practice;
public class Demo {
	public static void main(String []args){
		char a='d';
		boolean b;
		b=a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}//Explicit type casting
package practice;
public class Demo {
	public static void main(String []args){
		char a='d';
		boolean b;
		b=(boolean)a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

Compilation error for type casting.

Conclusion:

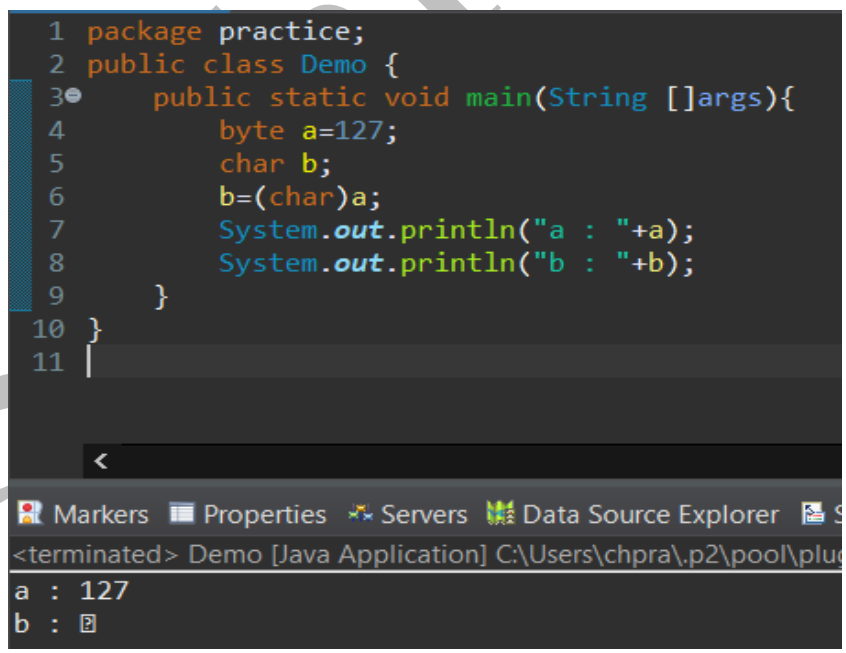
char-to-boolean conversion is not possible.

Conversion of data from byte-to-char datatype:

Program:

```
package practice;
public class Demo {
	public static void main(String []args){
		byte a=127;
		char b;
		b=(char)a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:



Conclusion:

byte-to-char conversion is done through explicit type casting.

Conversion of data from byte-to-byte datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                byte a=126;
                byte b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
1  package practice;
2  public class Demo {
3      public static void main(String []args){
4          byte a=126;
5          byte b;
6          b=a;
7          System.out.println("a : "+a);
8          System.out.println("b : "+b);
9      }
10 }
11
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plug
a : 126
b : 126
```

Conclusion:

byte-to-byte conversion is not required.

Conversion of data from byte-to-short datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                byte a=125;
                short b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
package practice;
public class Demo {
    public static void main(String []args){
        byte a=125;
        short b;
        b=a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer Sn

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plugi
a : 125
b : 125
```

Conclusion:

byte-to-short conversion is done through implicit type casting.

Conversion of data from byte-to-int datatype:

Program:

```
package practice;
public class Demo {
    public static void main(String []args){
        byte a=124;
        int b;
        b=a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Output:

```
package practice;
public class Demo {
    public static void main(String []args){
        byte a=124;
        int b;
        b=a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plug
a : 124
b : 124
```

Conclusion:

byte-to-int conversion is done through implicit type casting.

Conversion of data from byte-to-long datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                byte a=123;
                long b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
1  package practice;
2  public class Demo {
3      public static void main(String []args){
4          byte a=123;
5          long b;
6          b=a;
7          System.out.println("a : "+a);
8          System.out.println("b : "+b);
9      }
10 }
11
```

Markers Properties Servers Data Source Explorer Sni

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plugin
a : 123
b : 123
```

Conclusion:

byte-to-long conversion is done through implicit type casting.

Conversion of data from byte-to-float datatype:

Program:

```
package practice;
public class Demo {
	public static void main(String []args){
		byte a=-128;
		float b;
		b=a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

```
1  package practice;
2  public class Demo {
3      public static void main(String []args){
4          byte a=-128;
5          float b;
6          b=a;
7          System.out.println("a : "+a);
8          System.out.println("b : "+b);
9      }
10 }
11
```

Markers Properties Servers Data Source Explorer S

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plug
a : -128
b : -128.0
```

Conclusion:

byte-to-float conversion is done through implicit type casting.

Conversion of data from byte-to-double datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                byte a=-120;
                double b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
1  package practice;
2  public class Demo {
3      public static void main(String []args){
4          byte a=-120;
5          double b;
6          b=a;
7          System.out.println("a : "+a);
8          System.out.println("b : "+b);
9      }
10 }
11
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plu
a : -120
b : -120.0
```

Conclusion:

byte-to-double conversion is done through implicit type casting.

Conversion of data from byte-to-boolean datatype:

Program://Implicit type casting

```
package practice;
public class Demo {
	public static void main(String []args){
		byte a=-120;
		boolean b;
		b=a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}//Explicit type casting
package practice;
public class Demo {
	public static void main(String []args){
		byte a=-120;
		boolean b;
		b=(boolean)a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

Compilation error for type casting.

Conclusion:

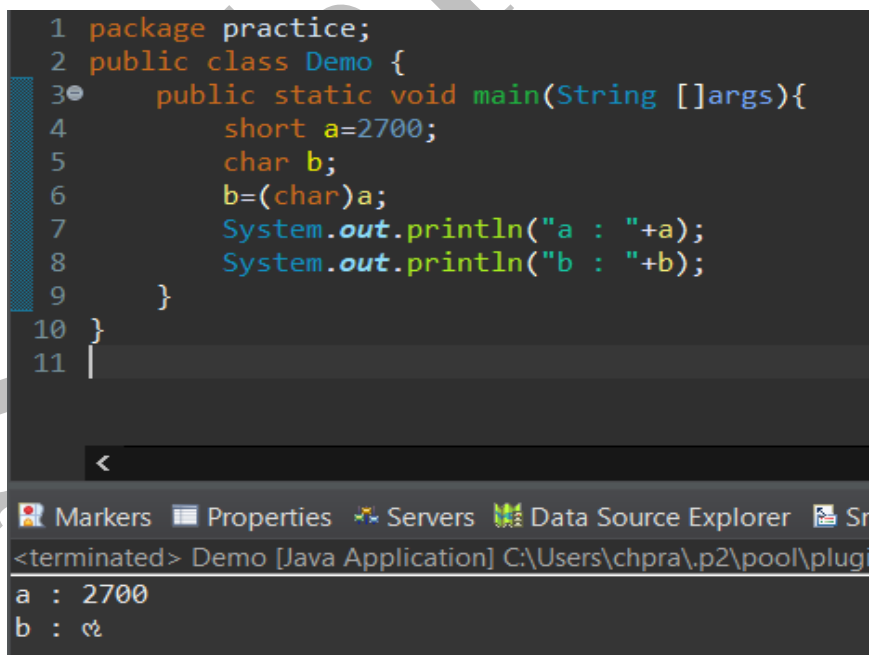
byte-to-boolean conversion is not possible.

Conversion of data from short-to-char datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                short a=2700;
                char b;
                b=(char)a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:



Conclusion:

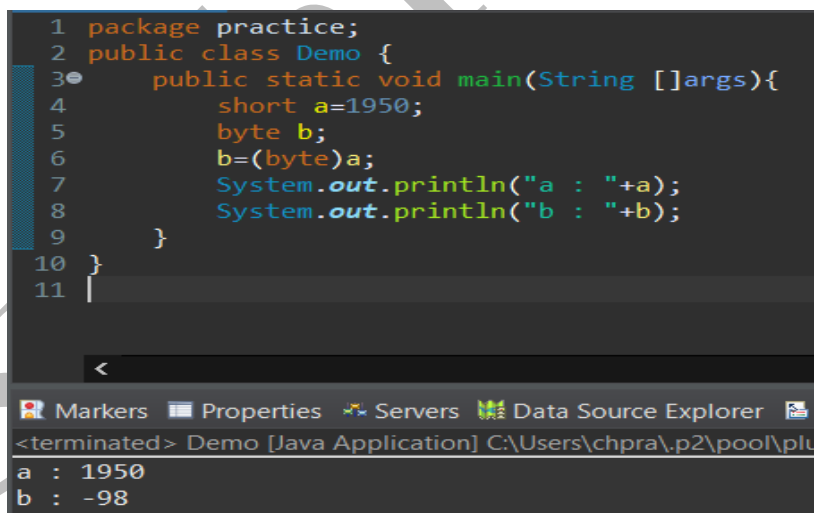
short-to-char conversion is done through explicit type casting.

Conversion of data from short-to-byte datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                short a=1950;
                byte b;
                b=(byte)a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:



Conclusion:

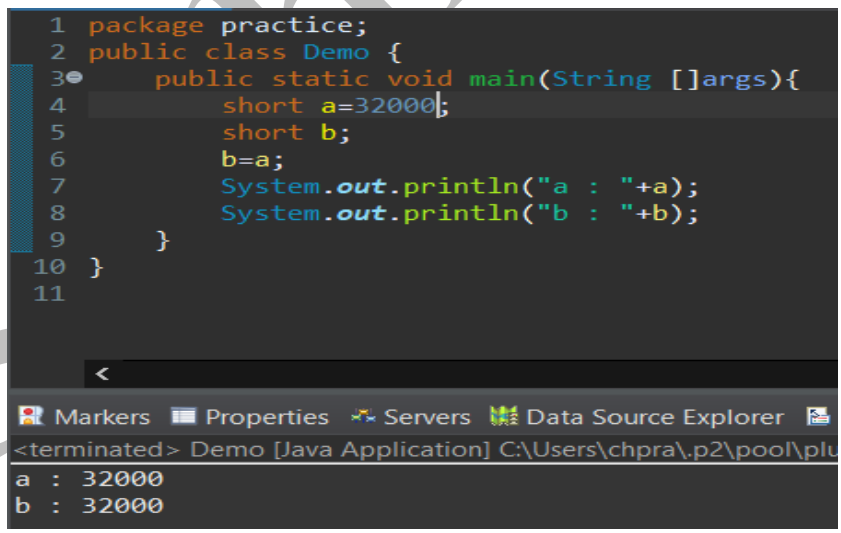
short-to-byte conversion is done through explicit type casting.

Conversion of data from short-to-short datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                short a=32000;
                short b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:



Conclusion:

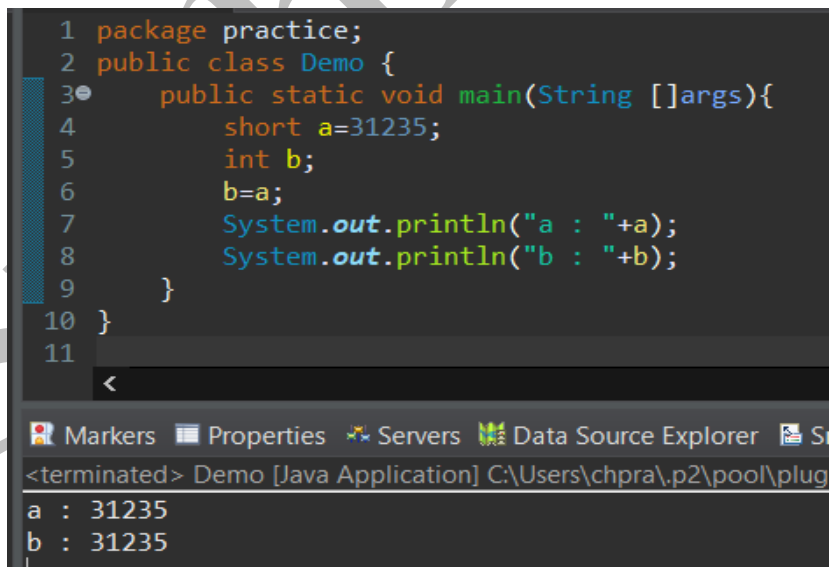
short-to-short conversion is not required.

Conversion of data from short-to-int datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                short a=31235;
                int b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:



Conclusion:

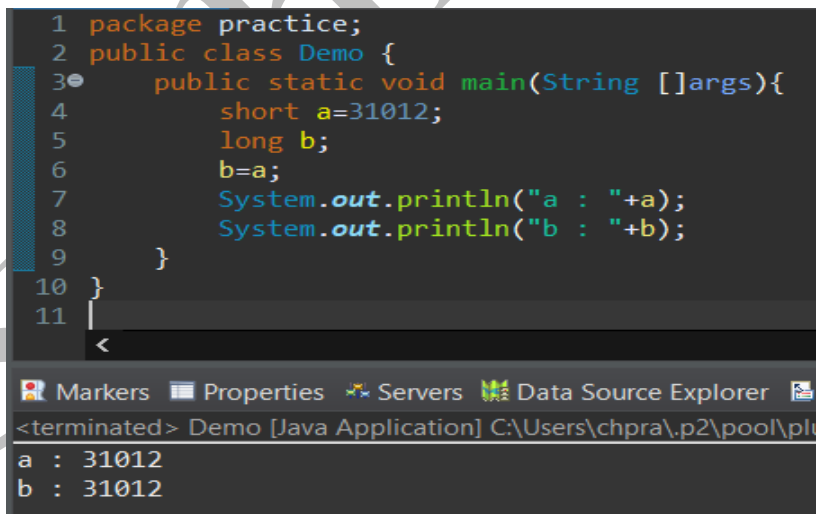
short-to-int conversion is done through implicit type casting.

Conversion of data from short-to-long datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                short a=31012;
                long b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:



Conclusion:

short-to-long conversion is done through implicit type casting.

Conversion of data from short-to-float datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                short a=-32001;
                float b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
1  package practice;
2  public class Demo {
3      public static void main(String []args){
4          short a=-32001;
5          float b;
6          b=a;
7          System.out.println("a : "+a);
8          System.out.println("b : "+b);
9      }
10 }
11
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\pl
a : -32001
b : -32001.0
```

Conclusion:

short-to-float conversion is done through implicit type casting.

Conversion of data from short-to-double datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                short a=-22601;
                double b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
package practice;
public class Demo {
    public static void main(String []args){
        short a=-22601;
        double b;
        b=a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plu
a : -22601
b : -22601.0
```

Conclusion:

short-to-double conversion is done through implicit type casting.

Conversion of data from short-to-boolean datatype:

Program://Implicit type casting

```
package practice;
public class Demo {
	public static void main(String []args){
		short a=3200;
		boolean b;
		b=a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}//Explicit type casting
package practice;
public class Demo {
	public static void main(String []args){
		short a=3200;
		boolean b;
		b=(boolean)a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

Compilation error for type casting.

Conclusion:

short-to-boolean conversion is not possible.

Conversion of data from int-to-char datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                int a=2000;
                char b;
                b=(char)a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
package practice;
public class Demo {
    public static void main(String []args){
        int a=2000;
        char b;
        b=(char)a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer S

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plug
a : 2000
b : ♀
```

Conclusion:

int-to-char conversion is done through explicit type casting.

Conversion of data from int-to-byte datatype:

Program:

```
package practice;
public class Demo {
	public static void main(String []args){
		int a=343001201;
		byte b;
		b=(byte)a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

```
1  package practice;
2  public class Demo {
3      public static void main(String []args){
4          int a=343001201;
5          byte b;
6          b=(byte)a;
7          System.out.println("a : "+a);
8          System.out.println("b : "+b);
9      }
10 }
11
```

Markers Properties Servers Data Source Explorer Sn

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plugi
a : 343001201
b : 113
```

Conclusion:

int-to-byte conversion is done through explicit type casting.

Conversion of data from int-to-short datatype:

Program:

```
package practice;
public class Demo {
	public static void main(String []args){
		int a=3430012;
		short b;
		b=(short)a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

```
package practice;
public class Demo {
    public static void main(String []args){
        int a=3430012;
        short b;
        b=(short)a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plu
a : 3430012
b : 22140
```

Conclusion:

int-to-short conversion is done through explicit type casting.

Conversion of data from int-to-int datatype:

Program:

```
package practice;
public class Demo {
	public static void main(String []args){
		int a=393192431;
		int b;
		b=a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

```
package practice;
public class Demo {
    public static void main(String []args){
        int a=393192431;
        int b;
        b=a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer Sn

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plugi
a : 393192431
b : 393192431
```

Conclusion:

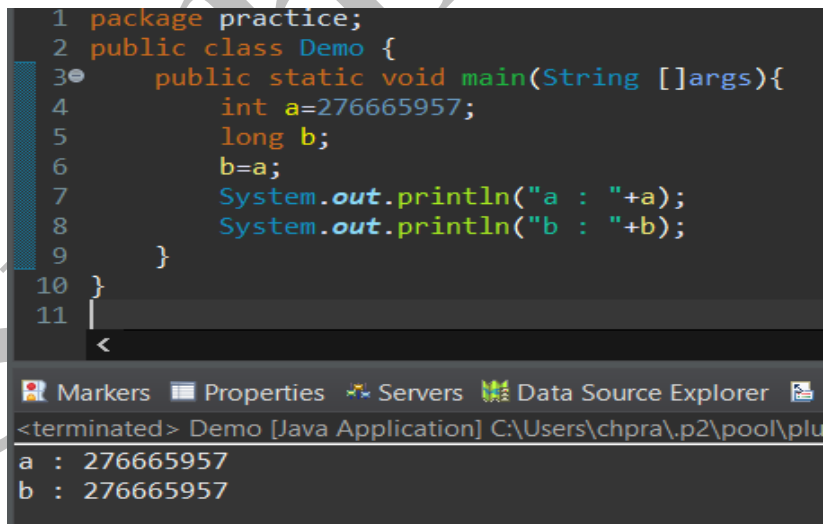
int-to-int conversion is not required.

Conversion of data from int-to-long datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                int a=276665957;
                long b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:



Conclusion:

int-to-long conversion is done through implicit type casting.

Conversion of data from int-to-float datatype:

Program:

```
package practice;
public class Demo {
	public static void main(String []args){
		int a=-279865957;
		float b;
		b=a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

```
1  package practice;
2  public class Demo {
3      public static void main(String []args){
4          int a=-279865957;
5          float b;
6          b=a;
7          System.out.println("a : "+a);
8          System.out.println("b : "+b);
9      }
10 }
11
```

Markers Properties Servers Data Source Explorer S

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plug
a : -279865957
b : -2.79865952E8
```

Conclusion:

int-to-float conversion is done through implicit type casting.

Conversion of data from int-to-double datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                int a=-1798657;
                double b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
1  package practice;
2  public class Demo {
3      public static void main(String []args){
4          int a=-1798657;
5          double b;
6          b=a;
7          System.out.println("a : "+a);
8          System.out.println("b : "+b);
9      }
10 }
11
```

Markers Properties Servers Data Source Explorer S

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plug
a : -1798657
b : -1798657.0
```

Conclusion:

int-to-double conversion is done through implicit type casting.

Conversion of data from int-to-boolean datatype:

Program://Implicit type casting

```
package practice;
public class Demo {
	public static void main(String []args){
		int a=1798657;
		boolean b;
		b=a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}//Explicit type casting
package practice;
public class Demo {
	public static void main(String []args){
		int a=1798657;
		boolean b;
		b=(boolean)a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

Compilation error for type casting.

Conclusion:

int-to-boolean conversion is not possible.

Conversion of data from long-to-char datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                long a=7000998743574l;
                char b;
                b=(char)a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
package practice;
public class Demo {
    public static void main(String []args){
        long a=7000998743574l;
        char b;
        b=(char)a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plug
a : 7000998743574
b : ฦ
```

Conclusion:

long-to-char conversion is done through explicit type casting.

Conversion of data from long-to-byte datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                long a=71232454l;
                byte b;
                b=(byte)a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
1  package practice;
2  public class Demo {
3      public static void main(String []args){
4          long a=71232454l;
5          byte b;
6          b=(byte)a;
7          System.out.println("a : "+a);
8          System.out.println("b : "+b);
9      }
10 }
11
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\pl
a : 71232454
b : -58
```

Conclusion:

long-to-byte conversion is done through explicit type casting.

Conversion of data from long-to-short datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                long a=-871232454l;
                short b;
                b=(short)a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
package practice;
public class Demo {
    public static void main(String []args){
        long a=-871232454l;
        short b;
        b=(short)a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plug
a : -871232454
b : 3130
```

Conclusion:

long-to-short conversion is done through explicit type casting.

Conversion of data from long-to-int datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                long a=12871232454l;
                int b;
                b=(int)a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
1  package practice;
2  public class Demo {
3      public static void main(String []args){
4          long a=12871232454l;
5          int b;
6          b=(int)a;
7          System.out.println("a : "+a);
8          System.out.println("b : "+b);
9      }
10 }
11
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plu
a : 12871232454
b : -13669434
```

Conclusion:

long-to-int conversion is done through explicit type casting.

Conversion of data from long-to-long datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                long a=7812871232454l;
                long b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
package practice;
public class Demo {
    public static void main(String []args){
        long a=7812871232454l;
        long b;
        b=a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plu
a : 7812871232454
b : 7812871232454
```

Conclusion:

long-to-long conversion is not required.

Conversion of data from long-to-float datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                long a=27812871232454l;
                float b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
package practice;
public class Demo {
    public static void main(String []args){
        long a=27812871232454l;
        float b;
        b=a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plug
a : 27812871232454
b : 2.78128702E13
```

Conclusion:

long-to-float conversion is done through implicit type casting.

Conversion of data from long-to-double datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                long a=51278128712324l;
                double b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
package practice;
public class Demo {
    public static void main(String []args){
        long a=51278128712324l;
        double b;
        b=a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plu
a : 51278128712324
b : 5.1278128712324E13
```

Conclusion:

long-to-double conversion is done through implicit type casting.

Conversion of data from long-to-boolean datatype:

Program://Implicit type casting

```
package practice;
public class Demo {
	public static void main(String []args){
		long a=751278128712324l;
		boolean b;
		b=a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}//Explicit type casting
package practice;
public class Demo {
	public static void main(String []args){
		long a=751278128712324l;
		boolean b;
		b=(boolean)a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

Compilation error for type casting.

Conclusion:

int-to-boolean conversion is not possible.

Conversion of data from float-to-char datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                float a=75127812f;
                char b;
                b=(char)a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
1  package practice;
2  public class Demo {
3      public static void main(String []args){
4          float a=75127812f;
5          char b;
6          b=(char)a;
7          System.out.println("a : "+a);
8          System.out.println("b : "+b);
9      }
10 }
11
```

Markers Properties Servers Data Source Explorer S

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plug
a : 7.5127808E7
b : 时
```

Conclusion:

float-to-char conversion is done through explicit type casting.

Conversion of data from float-to-byte datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                float a=10865127812f;
                byte b;
                b=(byte)a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
1  package practice;
2  public class Demo {
3      public static void main(String []args){
4          float a=10865127812f;
5          byte b;
6          b=(byte)a;
7          System.out.println("a : "+a);
8          System.out.println("b : "+b);
9      }
10 }
11
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plu
a : 1.08651274E10
b : -1
```

Conclusion:

float-to-byte conversion is done through explicit type casting.

Conversion of data from float-to-short datatype:

Program:

```
package practice;
public class Demo {
	public static void main(String []args){
		float a=-965127812f;
		short b;
		b=(short)a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

```
package practice;
public class Demo {
    public static void main(String []args){
        float a=-965127812f;
        short b;
        b=(short)a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plug
a : -9.6512781E8
b : 20864
```

Conclusion:

float-to-short conversion is done through explicit type casting.

Conversion of data from float-to-int datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                float a=-10965127812f;
                int b;
                b=(int)a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
1  package practice;
2  public class Demo {
3      public static void main(String []args){
4          float a=-10965127812f;
5          int b;
6          b=(int)a;
7          System.out.println("a : "+a);
8          System.out.println("b : "+b);
9      }
10 }
11
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plug
a : -1.09651282E10
b : -2147483648
```

Conclusion:

float-to-int conversion is done through explicit type casting.

Conversion of data from float-to-long datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                float a=9010965127812f;
                long b;
                b=(long)a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
package practice;
public class Demo {
    public static void main(String []args){
        float a=9010965127812f;
        long b;
        b=(long)a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\pl
a : 9.0109651E12
b : 9010965118976
```

Conclusion:

float-to-long conversion is done through explicit type casting.

Conversion of data from float-to-float datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                float a=89010965127812f;
                float b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
package practice;
public class Demo {
    public static void main(String []args){
        float a=89010965127812f;
        float b;
        b=a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plu
a : 8.9010966E13
b : 8.9010966E13
```

Conclusion:

float-to-float conversion is not required.

Conversion of data from float-to-double datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                float a=2189010965127812f;
                double b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
1  package practice;
2  public class Demo {
3      public static void main(String []args){
4          float a=2189010965127812f;
5          double b;
6          b=a;
7          System.out.println("a : "+a);
8          System.out.println("b : "+b);
9      }
10 }
11
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plu
a : 2.18901102E15
b : 2.189011015696384E15
```

Conclusion:

float-to-double conversion is done through implicit type casting.

Conversion of data from float-to-boolean datatype:

Program://Implicit type casting

```
package practice;
public class Demo {
	public static void main(String []args){
		float a=10965127812f;
		boolean b;
		b=a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}//Explicit type casting
package practice;
public class Demo {
	public static void main(String []args){
		float a=10965127812f;
		boolean b;
		b=(boolean)a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

Compilation error for type casting.

Conclusion:

float-to-boolean conversion is not possible.

Conversion of data from double-to-char datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                double a=1296512782;
                char b;
                b=(char)a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
package practice;
public class Demo {
    public static void main(String []args){
        double a=1296512782;
        char b;
        b=(char)a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plu
a : 1.296512782E9
b : 嫒
```

Conclusion:

double-to-char conversion is done through explicit type casting.

Conversion of data from double-to-byte datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                double a=53.769526555953525;
                byte b;
                b=(byte)a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
1  package practice;
2  public class Demo {
3      public static void main(String []args){
4          double a=53.769526555953525;
5          byte b;
6          b=(byte)a;
7          System.out.println("a : "+a);
8          System.out.println("b : "+b);
9      }
10 }
11
```

Markers Properties Servers Data Source Explorer S

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plug
a : 53.769526555953526
b : 53
```

Conclusion:

double-to-byte conversion is done through explicit type casting.

Conversion of data from double-to-short datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                double a=-654.769512655593;
                short b;
                b=(short)a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
package practice;
public class Demo {
    public static void main(String []args){
        double a=-654.769512655593;
        short b;
        b=(short)a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer Sn

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plug
a : -654.769512655593
b : -654
```

Conclusion:

double-to-short conversion is done through explicit type casting.

Conversion of data from double-to-int datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                double a=1769.088765547;
                int b;
                b=(int)a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
1  package practice;
2  public class Demo {
3      public static void main(String []args){
4          double a=1769.088765547;
5          int b;
6          b=(int)a;
7          System.out.println("a : "+a);
8          System.out.println("b : "+b);
9      }
10 }
11
```

Markers Properties Servers Data Source Explorer Snippet

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plugins\org
a : 1769.088765547
b : 1769
```

Conclusion:

double-to-int conversion is done through explicit type casting.

Conversion of data from double-to-long datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                double a=7169.7588765547;
                long b;
                b=(long)a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
1  package practice;
2  public class Demo {
3      public static void main(String []args){
4          double a=7169.7588765547;
5          long b;
6          b=(long)a;
7          System.out.println("a : "+a);
8          System.out.println("b : "+b);
9      }
10 }
11
```

Markers Properties Servers Data Source Explorer S

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plug
a : 7169.7588765547
b : 7169
```

Conclusion:

double-to-long conversion is done through explicit type casting.

Conversion of data from double-to-float datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                double a=91169.7588765547;
                float b;
                b=(float)a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
package practice;
public class Demo {
    public static void main(String []args){
        double a=91169.7588765547;
        float b;
        b=(float)a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer Snippe

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plugins\o
a : 91169.7588765547
b : 91169.76
```

Conclusion:

double-to-float conversion is done through explicit type casting.

Conversion of data from double-to-double datatype:

Program:

```
package practice;
public class Demo {
	public static void main(String []args){
		double a=391169.7588765547;
		double b;
		b=a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

```
package practice;
public class Demo {
    public static void main(String []args){
        double a=391169.7588765547;
        double b;
        b=a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer S

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plug
a : 391169.7588765547
b : 391169.7588765547
```

Conclusion:

double-to-double conversion is not required.

Conversion of data from double-to-boolean datatype:

Program://Implicit type casting

```
package practice;
public class Demo {
	public static void main(String []args){
		double a=1096512782;
		boolean b;
		b=a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}//Explicit type casting
package practice;
public class Demo {
	public static void main(String []args){
		double a=1096512782;
		boolean b;
		b=(boolean)a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

Compilation error for type casting.

Conclusion:

double-to-boolean conversion is not possible.

Conversion of data from boolean-to-char datatype:

Program:

```
package practice;
public class Demo {
	public static void main(String []args){
		boolean a=true;
		char b;
		b=a;
		// explicit type casting
		b=(char)a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

Compilation error for type casting.

Conclusion:

boolean-to-char conversion is not possible.

Conversion of data from boolean-to-byte datatype:

Program:

```
package practice;
public class Demo {
	public static void main(String []args){
		boolean a=true;
		byte b;
		b=a;
		// explicit type casting
		b=(byte)a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

Compilation error for type casting.

Conclusion:

boolean-to-byte conversion is not possible.

Conversion of data from boolean-to-short datatype:

Program:

```
package practice;
public class Demo {
	public static void main(String []args){
		boolean a=true;
		short b;
		b=a;
		// explicit type casting
		b=(short)a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

Compilation error for type casting.

Conclusion:

boolean-to-short conversion is not possible.

Conversion of data from boolean-to-int datatype:

Program:

```
package practice;
public class Demo {
	public static void main(String []args){
		boolean a=true;
		int b;
		b=a;
		// explicit type casting
		b=(int)a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

Compilation error for type casting.

Conclusion:

boolean-to-int conversion is not possible.

Conversion of data from boolean-to-long datatype:

Program:

```
package practice;
public class Demo {
	public static void main(String []args){
		boolean a=true;
		long b;
		b=a;
		// explicit type casting
		b=(long)a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

Compilation error for type casting.

Conclusion:

boolean-to-long conversion is not possible.

Conversion of data from boolean-to-float datatype:

Program:

```
package practice;
public class Demo {
	public static void main(String []args){
		boolean a=true;
		float b;
		b=a;
		// explicit type casting
		b=(float)a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

Compilation error for type casting.

Conclusion:

boolean-to-float conversion is not possible.

Conversion of data from boolean-to-double datatype:

Program:

```
package practice;
public class Demo {
	public static void main(String []args){
		boolean a=true;
		double b;
		b=a;
		// explicit type casting
		b=(double)a;
		System.out.println("a : "+a);
		System.out.println("b : "+b);
	}
}
```

Output:

Compilation error for type casting.

Conclusion:

boolean-to-double conversion is not possible.

Conversion of data from boolean-to-boolean datatype:

Program:

```
package practice;
public class Demo {
        public static void main(String []args){
                boolean a=true;
                boolean b;
                b=a;
                System.out.println("a : "+a);
                System.out.println("b : "+b);
        }
}
```

Output:

```
package practice;
public class Demo {
    public static void main(String []args){
        boolean a=true;
        boolean b;
        b=a;
        System.out.println("a : "+a);
        System.out.println("b : "+b);
    }
}
```

Markers Properties Servers Data Source Explorer Sni

```
<terminated> Demo [Java Application] C:\Users\chpra\.p2\pool\plugir
a : true
b : true
```

Conclusion:

boolean-to-boolean conversion is not required.

char→byte→short→int→long→float→double→boolean

---------- implicit conversion

boolean→double→float→long→int→short→byte→char

----------explicit conversion

	char	byte	short	Int	long	Float	double	boolean
char	CNR	✓ Explicit	✓ Explicit	✓ Implicit	✓ Implicit	✓ Implicit	✓ Implicit	✗ Not possible
byte	✓ Explicit	CNR	✓ Implicit	✓ Implicit	✓ Implicit	✓ Implicit	✓ Implicit	✗ Not possible
short	✓ Explicit	✓ Explicit	CNR	✓ Implicit	✓ Implicit	✓ Implicit	✓ Implicit	✗ Not possible
int	✓ Explicit	✓ Explicit	✓ Explicit	CNR	✓ Implicit	✓ Implicit	✓ Implicit	✗ Not possible
long	✓ Explicit	✓ Explicit	✓ Explicit	✓ Explicit	CNR	✓ Implicit	✓ Implicit	✗ Not possible
float	✓ Explicit	✓ Explicit	✓ Explicit	✓ Explicit	✓ Explicit	CNR	✓ Implicit	✗ Not possible
double	✓ Explicit	✓ Explicit	✓ Explicit	✓ Explicit	✓ Explicit	✓ Explicit	CNR	Not possible
boolean	✗ Not possible	✗ Not possible	✗ Not possible	✗ Not possible	✗ Not possible	✗ Not possible	✗ Not possible	CNR

CNR: Conversion not required