```java
package com.publicept;

public class Main {

    public static void main(String[] args) {

        /**
         * int
         */
        // int = width of 32 bits
        int myValue = 10_000;

        int myMinIntValue = Integer.MIN_VALUE;
        int myMaxIntValue = Integer.MAX_VALUE;
        System.out.println("Integer min. Value = " + myMinIntValue); // -2147483648
        System.out.println("Integer max. Value = " + myMaxIntValue); // 2147483647
        System.out.println("Busted MAX Value (\"Overflow\") = " + (myMaxIntValue + 1)); // -2147483648
        System.out.println("Busted MIN Value (\"Underflow\") = " + (myMinIntValue - 1)); // 2147483647

        int myMaxIntTest = 2_147_483_647;   // easy to read with underscore (Java 7+)


        /**
         * byte
         */
        // byte = width of 8
        byte myMinByteValue = Byte.MIN_VALUE;
        byte myMaxByteValue = Byte.MAX_VALUE;
        System.out.println("Byte MIN Value = " + myMinByteValue); // -128
        System.out.println("Byte MAX Value = " + myMaxByteValue); // 127


        /**
         * short
         */
        // short = width of 16
        short myMinShortValue = Short.MIN_VALUE;
        short myMaxShortValue = Short.MAX_VALUE;
        System.out.println("Short MIN Value = " + myMinShortValue); // -32768
        System.out.println("Short MAX Value = " + myMaxShortValue); // 32767


        /**
         * long
         */
        // long = width of 64
        //long myLongValue = 100; // technically wrong
        long myLongValue = 100L; // Upper-case L at the end needed to tell the computer that it's a long value
        long myMinLongValue = Long.MIN_VALUE;
        long myMaxLongValue = Long.MAX_VALUE;
        System.out.println("Long MIN Value = " + myMinLongValue); // -9_223_372_036_854_775_808
        System.out.println("Long MAX Value = " + myMaxLongValue); // 9_223_372_036_854_775_807

        // error "Integer number too large", as default number is Integer in Java!
        // long bigLongLiteralValue = 2_147_483_647_234; // assign int value to long var
        long bigLongLiteralValue = 2_147_483_647_234L; // assign long with Literal [L]

        System.out.println("Literal big long value = " + bigLongLiteralValue);


        /**
         * Casting
         */
        // short bigShortLiteralValue = 32768; // error: "incompatible types: required short, found int"
        short bigShortLiteralValue = 32767;

        int myTotal = (myMinIntValue / 2);

        // byte myNewByteValue = (myMinByteValue / 2); // error: expressions in parenthesis are treated as int in Java
        byte myNewByteValue = (byte) (myMinByteValue / 2);  // casting of byte within parenthesis

        short myNewShortValue = (short) (myMinShortValue / 2);  // casting again
        // advice always take an integer instead of the real need of another data type

    }
}
```