

EJERCICIO U1_B3_E1.

```
class App{
    public static void main(String[] args){
        System.out.println("Tipo\tMin.\t\tMax.");
        System.out.println("int\t"+Integer.MIN_VALUE+"\t"+Integer.MAX_VALUE);
        System.out.println("byte\t"+Byte.MIN_VALUE+"\t\t"+Byte.MAX_VALUE);
        System.out.println("short\t"+Short.MIN_VALUE+"\t\t"+Short.MAX_VALUE);
        System.out.println("float\t"+Float.MIN_VALUE+"\t\t"+Float.MAX_VALUE);
        System.out.println("double\t"+Double.MIN_VALUE+"\t\t"+Double.MAX_VALUE);
    }
}
```

EJERCICIO U1_B3_E2.

El 8 no es símbolo del código octal. Octal usa dígitos 0,1,2,3,4,5,6,7. Si comenzamos con 0 estamos indicando a java que vamos a escribir en octal y 08 no es un número octal válido. Nos fijamos que 07 es ok pero 08.

```
jshell> int i
i ==> 0

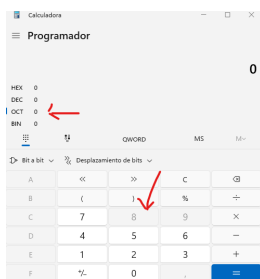
jshell> i=0123
i ==> 83

jshell> i=08
| Error:
| ';' expected
| i=08
|   ^

jshell> i=07
i ==> 7

jshell>
```

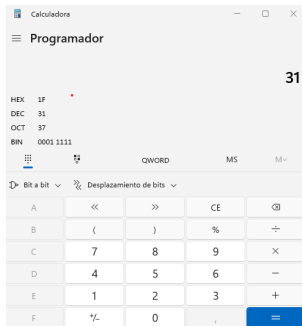
En la calculadora windows que nos permite escribir en octal, observamos que si hacemos clic en OCT está deshabilitado el dígito 8



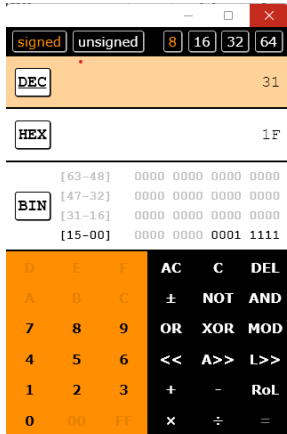
También podríamos comprobar que si hacemos clic en HEX aparecen el teclado las letras de la A a la F

EJERCICIO U1_B3_E3.

En calculadora windows



en calculadora chrome



En chrome no viene octal, pero para observar los cast del próximo boletín nos va a gustar más la calculadora de windows

EJERCICIO U1_B3_E4.

Aunque sólo sea por 1, nos pasamos de rango. Consulta el máximo en el ejercicio 1

EJERCICIO U1_B3_E5

128 está dentro del rango int pero fuera del rango byte. Consulta rangos en ejercicio 1.

EJERCICIO U1_B3_E6.

10.0 es una literal double. Es equivalente al literal 10.0d. No se puede asignar un literal double a una variable float. El literal float equivalente es 10.0f. Si añadimos una f, desaparece el error.

```
class App{
    public static void main(String[] args) {
        float miFloat=10.0f;
        double miDouble=10.0;
    }
}
```

EJERCICIO U1_B3_E7.

Dentro de ""(comillas dobles) o de " (comillas simples) la barra es un carácter especial que advierte que el siguiente carácter tiene un uso especial. A la combinación de estos dos caracteres le llamamos secuencia de escape.

Pero sólo unos cuantos caracteres se pueden usar para hacer secuencia de escape, la p por ejemplo no vale para formar secuencia de escape.

EJERCICIO U1_B3_E8.

porque la combinación \n es una secuencia de escape y tiene un significado especial, en este caso origina un salto de línea.

EJERCICIO U1_B3_E9.

```
class App{
    public static void main(String[] args){
        System.out.println("Me gusta mucho la barra \\ y las comillas \");
    }
}
```

Para anular el sentido sintáctico especial de \ y " dentro de String, las antecede de \ y se convierten en caracteres de texto normales imprimibles.

EJERCICIO U1_B3_E10.

Es el valor 65. Usando 65 obtengo el mismo resultado.

```
class App{
    public static void main(String[] args){
        char miLetra_A=65;
        System.out.println(miLetra_A);
    }
}
```

La relación int/char tiene una serie de matices que se iran abordando poco a poco.

EJERCICIO U1_B3_E11.

```
class App{
    public static void main(String[] args){
        String nombre="Juán López Salvado";
        char sexo='v';
        short edad=49;//hay casos de edad de más de 127
        float peso=70.8f;
        System.out.println("nombre: "+ nombre);
        System.out.println("sexo: "+ sexo);
        System.out.println("edad: "+ edad);
        System.out.println("peso: "+ peso);
    }
}
```

EJERCICIO U1_B3_E12.

178: tipo int, el valor está especificado en base decimal.

2L: Tipo long, el valor está especificado en base decimal.

077L: tipo long, el valor está especificado en octal.

0xBAACL: tipo long, el valor está especificado en hexadecimal.

37.266D: tipo double, en base 10

87.363F: tipo float, en base 10

'c': tipo char

'\t': tipo char

true: tipo boolean

false: tipo boolean

'\u00E1': tipo char

"hola": no es un tipo primitivo, es un objeto String