

Ejercicio U3_B1_E1

```
import java.util.Scanner;
public class Unidad3{
    public static void main(String[] args) {
        Scanner entrada=new Scanner(System.in);
        int x,y;
        System.out.println("Teclea número entero x");
        x=entrada.nextInt();
        System.out.println("Teclea número entero y");
        y=entrada.nextInt();
        if (x== 10){
            System.out.println("x vale 10");
            System.out.println("x + y vale: " + (x+y));
        }else{//quedaría más consistente usar llaves también en este else. Así "rechina"
            System.out.println("x NO vale 10");
        }
    }
}
```

Ejercicio U3_B1_E2

```
import java.util.Scanner;
public class U3_B1_E2{
    public static void main(String[] args) {
        Scanner entrada=new Scanner(System.in);
        int x,y;
        System.out.println("Teclea número entero x");
        x=entrada.nextInt();
        System.out.println("Teclea número entero y");
        y=entrada.nextInt();
        if (x== 10){
            System.out.println("x vale 10");
            System.out.println("x + y vale: " + (x+y));
        }else{
            System.out.println("x NO vale 10");
            System.out.println("x - y vale: " + (x-y));
        }
    }
}
```

Ejercicio U3_B1_E3

La variable `y` está definida en el bloque exterior y se puede utilizar en el bloque interior. Cuando en el bloque interior se intenta definir una variable con el mismo nombre el compilador nos advierte de la "colisión de nombres". La solución sería utilizar otro nombre de variable en el exterior o interior,

Observa que la colisión de nombres anterior con el correspondiente error de compilación, ocurre en el caso anterior porque ambas variables son variables locales. Por el contrario, si repasas lo visto al respecto de *this* en la unidad 3, no se produce error de compilación cuando coincide el nombre de una variable local con una variable miembro, simplemente, en este caso, la variable local prima en su ámbito y oculta a la variable miembro en dicho ámbito. La solución a esto era utilizar *this* para referenciar a la variable miembro (o cambiar uno de los nombres).

Ejercicio U3_B1_E4

A una de las variables hay que cambiarle de nombre por ejemplo a la `y` externa la llamamos `z`

```
class Unidad3 {
    public static void main(String[] args) {
        int x,z;
        x=10;
        z=13;
        if (x== 10){
            int y = 20;
            System.out.println("x:" + x + " y: " + y);
            x = y * 2;
        }
    }
}
```

```

        z= ++y;
    }
    System.out.println("x:" + x);
    System.out.println("z:" + z);
}
}

```

Ejercicio U3_B1_E5

```

class Unidad3{
    public static void main(String args[]){
        int x;
        x=3;
        if (x==1)
            System.out.println("x es uno");
        else if (x==2)
            System.out.println("x es dos");
        else if (x==3)
            System.out.println("x es tres");
        else if (x==4)
            System.out.println("x es cuatro");
        else if (x==5)
            System.out.println("x es cinco");
        else
            System.out.println("x no se encuentra entre uno y cinco");
    }
}

```

o con las llaves del if then por si queremos añadir más instrucciones

```

class Unidad3 {

    public static void main(String args[]) {
        int x;
        x = 3;
        if (x == 1) {
            System.out.println("x es uno");
        } else if (x == 2) {
            System.out.println("x es dos");
        } else if (x == 3) {
            System.out.println("x es tres");
        } else if (x == 4) {
            System.out.println("x es cuatro");
        } else if (x == 5) {
            System.out.println("x es cinco");
        } else {
            System.out.println("x no se encuentra entre uno y cinco");
        }
    }
}

```

Ejercicio U3_B1_E6

```

import java.util.Scanner;

class Unidad3{

    public static void main(String[] args) {
        Scanner teclado= new Scanner(System.in);

        int a,b,s=5;

        System.out.println("a: ");
        a=teclado.nextInt();
    }
}

```

```

System.out.println("b: ");
b=teclado.nextInt();

if(a==0)
    if(b!=0)
        s=s+b;
    else
        s=s+a;
System.out.println("s: "+s);
}
}

```

Aunque no lo pide el enunciado, en este caso, el código se lee mejor con brackets

```

if(a==0){
    if(b!=0){
        s=s+b;
    }else{
        s=s+a;
    }
}
}

```

Ejercicio U3_B1_E7. Lo mejor es razonarlo y usar llaves. Observa que esto no es un caso de if anidado dentro de else si no if anidado dentro de if

```

import java.util.Scanner;
class Unidad3{
    public static void main(String[] args) {
        Scanner teclado= new Scanner(System.in);
        int a,b,s=5;
        System.out.println("a: ");
        a=teclado.nextInt();
        System.out.println("b: ");
        b=teclado.nextInt();

        if(a==0){
            if(b!=0){
                s=s+b;
            }else{
                s=s+a;
            }
        }
        System.out.println("s: "+s);
    }
}

```

La instrucción $s=s+a$ es superflua ya que sólo se ejecuta cuando a vale 0 con lo cual: $s+a$ es siempre $s+0$ y por tanto nos quedaría la instrucción $s=s$ que no aporta nada y se puede suprimir.

```

import java.util.Scanner;
class Unidad3 {

```

```

public static void main(String[] args) {
    Scanner teclado= new Scanner(System.in);
    int a,b,s=5;
    System.out.println("a: ");
    a=teclado.nextInt();
    System.out.println("b: ");
    b=teclado.nextInt();

    if(a==0)
        if(b!=0)
            s=s+b;
    System.out.println("s: "+s);
}
}

```

ahora observo que también sobra `if(b!=0)` ya que si realmente si `b` vale 0 aunque lo sume a `s` no altera el resultado. Por tanto queda:

```

import java.util.Scanner;
class Unidad3 {
    public static void main(String[] args) {
        Scanner teclado= new Scanner(System.in);
        int a,b,s=5;
        System.out.println("a: ");
        a=teclado.nextInt();
        System.out.println("b: ");
        b=teclado.nextInt();

        if(a==0)
            s=s+b;
        System.out.println("s: "+s);
    }
}

```

Observa que en cambio si en lugar de
`s=s+b;`
 la operación fuera
`s=s/b;`
 si que preciso el `if` para prevenir división por cero

Ejercicio U3_B1_E8.

```

import java.util.Scanner;
class Unidad3 {
    public static void main(String[] args) {
        Scanner teclado= new Scanner(System.in);
        int a,b,s=5;
        System.out.println("a: ");
        a=teclado.nextInt();
        System.out.println("b: ");
        b=teclado.nextInt();

        if(a==0)
            if(b!=0)
                s=s/b;
            else
                s++;
        System.out.println("s: "+s);
    }
}

```

```
}
```

Que es mejor ¿así, o como se escribió en el enunciado?. Lo mejor sería lo más connatural al problema a resolver, como el ejercicio no responde a ninguna situación real concreta no podemos saber qué es lo mejor. En cualquier caso, las dos soluciones serían correctas.

Ejercicio U3_B1_E9.

```
import java.util.Scanner;

class Unidad3 {
    public static void main(String[] args) {
        Scanner teclado= new Scanner(System.in);
        int a,b,s=5; //mejor, es cada variable en su línea pero existe esta posibilidad
        System.out.println("a: ");
        a=teclado.nextInt();
        System.out.println("b: ");
        b=teclado.nextInt();

        if(a==0)
            if(b==0)
                s++;
            else
                s+=b;
        else
            if(b==0)
                s=999+a;
            else
                s=888+a+b;
        System.out.println("s: "+s);
    }
}
```

Ejercicio U3_B1_E10.

el operador condicional se puede anidar

```
class Unidad3{
    public static void main(String args[]){
        int x;
        x=2;
        System.out.println(x==1?"x es uno": (x==2?" x es dos":"x es tres" ) );
    }
}
```

Ejercicio U3_B1_E11.

```
import java.util.Scanner;
class Unidad3 {
    public static void main(String[] args) {
        //suponemos 3 enteros distintos
        Scanner sc= new Scanner(System.in);
        System.out.println("Mete 3 números separados: ");
        int a=sc.nextInt();
        int b=sc.nextInt();
        int c=sc.nextInt();
        int menor=a;
        if(b<menor){
            menor=b;
        }
        if (c<menor){
            menor=c;
        }
        System.out.println("El menor: "+ menor);
    }
}
```

```
}
```

Ejercicio U3_B1_E12.

```
class Unidad3 {  
  
    static int max1(int a, int b, int c) {  
        if (a >= b && a >= c) {  
            return a;  
        }  
        if (b >= a && b >= c) {  
            return b;  
        }  
        return c;  
    }  
  
    static int max2(int a, int b, int c) {  
        int max;  
        if (a >= b && a >= c) { //el máximo es a  
            max=a;  
        }else{//el máximo es b o c  
            if(b>=c){  
                max=b;  
            }else{  
                max=c;  
            }  
        }  
  
        return max;  
    }  
  
    public static void main(String args[]) {  
  
        System.out.println(max1(4, 6, 5));  
        System.out.println(max1(9, 6, 5));  
        System.out.println(max1(1, 6, 88));  
  
        System.out.println(max2(4, 6, 5));  
        System.out.println(max2(9, 6, 5));  
        System.out.println(max2(1, 6, 88));  
  
    }  
}
```