

Ejercicios para practicar listas anidadas

1. Desarrolla una función **creaMatriz1** la cual recibe un entero n que representa el número de renglón y columnas. La función devuelve una lista que representa una matriz de nXn y cada localidad de la matriz tiene asignado el valor de -1.
2. Desarrolla una función **creaMatriz2** la cual recibe un entero n que representa el número de renglón y columnas. La función devuelve una lista que representa una matriz de nXn y cada localidad de la matriz tiene asignado el número de su respectiva columna.
3. Desarrolla una función **creaMatriz3** la cual recibe un entero n que representa el número de renglón y columnas. La función devuelve una lista que representa una matriz de nXn y cada localidad de la matriz tiene asignado el valor de su correspondiente renglón.
4. Desarrolla una función **creaMatriz4** la cual recibe un entero n que representa el número de renglón y columnas. La función devuelve una lista que representa una matriz de nXn y cada localidad de la matriz tiene asignado un número consecutivo a partir de uno. **Observa el ejemplo:**

1	5	9	13
2	6	10	14
3	7	11	15
4	8	12	16

5. Desarrolla una función **cuentaPares**, la cual recibe como parámetro una lista anidada de números y regresa cuántos números pares hay dentro de la lista.
6. Desarrolla una función **sumaMatriz**, que reciba como parámetro una lista anidada de números. El método regresa el resultado de sumar los valores de cada una de las localidades.
7. Desarrolla una función **cuentaPositivos**, que reciba como parámetro una lista anidada de números. El método regresa el número de valores mayores o iguales que cero almacenados en la lista.
8. Desarrolla una función **cambiaNegativos**, que reciba como parámetro una lista anidada de números. El método modifica la lista, asignando el valor de 0 a cada localidad que contenga un valor menor a cero.
9. Desarrolla una función **cuentaRepeticiones** que recibe como parámetro una lista anidada de números y un valor entero x. El método regresa el número de veces que se repite el valor de x en la lista.
10. Desarrolla una función **busca** que recibe como parámetro una lista anidada de números y un valor entero x. El método regresa el valor

true si el valor de x existe en una localidad de la matriz y **false** si no existe.

11. Desarrolla una función **sumaMayores5** que recibe como parámetro una lista anidada de números. El método deberá devolver la suma de todos aquellos números que haya dentro de la matriz que sean mayores a 5.
12. Desarrolla una función **cambiaCeros** que recibe como parámetro una lista anidada de números. El método deberá sustituir todos aquellos ceros que haya dentro de la matriz por el resultado de la suma de su número de renglón por su número de columna.