

---

## EJERCICIOS INTERFACES FUNCIONALES Y STREAM

---

### EJERCICIO 1 - PREDICADOS:

1. Predicado para números positivos: Verificar si un número entero es positivo.
2. Predicado para cadenas no vacías: Comprobar si una cadena de texto no está vacía.
3. Predicado para números pares: Determinar si un número entero es par.
4. Predicado para números mayores que un valor dado. Comprobar si un número entero es mayor que un valor específico.

### EJERCICIO 2 - CONSUMIDORES:

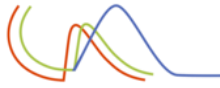
1. Consumidor para imprimir en mayúsculas: Tomar una cadena de texto y mostrarla en mayúsculas.
2. Consumidor para imprimir números y sus cuadrados: Tomar un número entero y mostrarlo junto con su cuadrado.
3. Consumidor para imprimir detalles de un producto: Tomar un objeto Producto y mostrar sus detalles como nombre, precio, etc.

### EJERCICIO 3 -PROVEEDORES:

1. Proveedor de números aleatorios: Generar un número aleatorio en el rango especificado.
2. Proveedor de una lista predefinida. Devolver una lista predefinida de elementos.
3. Proveedor de fecha y hora actual: Obtener la fecha y hora actual del sistema.
4. Proveedor de un nuevo objeto String vacío: Crear y devolver una cadena de texto vacía.

### EJERCICIO 4 - FUNCTION

1. Filtrado de elementos de una lista: Define una función que tome una lista de números como entrada y devuelva una lista que solo contenga los números pares. Utiliza la interfaz funcional Function para filtrar los elementos de la lista.
2. Cálculo de estadísticas: Escribe una función que tome una lista de números como entrada y devuelva un objeto Stats que contenga estadísticas como el mínimo, el máximo, la media de los números en la lista. Utiliza la interfaz funcional Function para realizar los cálculos necesarios.
3. Validación de contraseñas: Crea una función que tome una cadena que representa una contraseña y devuelva true si la contraseña cumple con ciertos criterios de validación, como tener al menos ocho caracteres, incluir al menos una letra mayúscula, una letra minúscula, un número y un carácter especial. Utiliza la interfaz funcional Function para representar cada criterio de validación como una función independiente y combina estas funciones para verificar la contraseña completa.



---

#### EJERCICIO 4 – ACCESO A BBDD / STREAM

Obtener los siguientes resultados desde la tabla de productos:

1. Obtener los productos cuya categoría sea 2
2. Obtener los productos que comiencen con N
3. Ordenar los productos de forma ascendente por nombre de producto
4. Obtener el producto con el precio máximo
5. Obtener el promedio del precio de producto
6. Obtener los empleados jubilados > 68
7. Obtener las categorías y la sumatoria de las cantidades de productos

#### EJERCICIO 5 - ACCESO A BBDD / STREAM

Obtener los siguientes resultados desde la tabla de productos:

1. Obtener las distintas ciudades de los empleados.

#### EJERCICIO 6 - ACCESO A BBDD / STREAM

Realizar acceso a BBDD Northwind a la tabla order\_details con un método para consultar todos los detalles de ordenes order\_details. Realizar los siguientes ejercicios aplicando interfaces funcionales y Streams.

2. Obtener los detalles de ordenes cuyo precio unitario sea mayor que 30.
3. Ordenar los productos de forma descendente por unit\_price
4. Obtener el detalle de ordenes cuyo precio es el mínimo
5. Obtener las estadísticas de detalle de ordenes sobre el atributo quantity
6. Obtener los productos y la suma de sus cantidades.