

# Introducción a la Algoritmia

Clase 11

<https://docs.python.org/3/reference/>  
<https://wiki.python.org.ar/aprendiendopython/>  
<https://docs.python.org/es/3/library/tkinter.html>

# Temas de la clase 11

Aplicaciones de las listas.

Inserción y eliminación de elementos.

# Interface Gráfica con tkinter

```
import tkinter as tk
```

```
# Crear la ventana principal
```

```
root = tk.Tk()
```

```
root.title("Mi primera aplicación")
```

```
# Definir el tamaño de la ventana
```

```
root.geometry("800x600")
```

```
# Iniciar el bucle principal de la aplicación
```

```
root.mainloop()
```

# Interface Gráfica con tkinter, botón para cerrar ventana

```
import tkinter as tk
# Función para cerrar la ventana
def cerrar():
    root.destroy()

root = tk.Tk()
root.title("Crud de artículos")
root.geometry("800x600")
# Crear un botón y colocarlo en la ventana
btn_cerrar = tk.Button(root, text="Cerrar", command=cerrar)
btn_cerrar.grid(row=10, column=8, pady=5)
# Iniciar el bucle principal de la aplicación
root.mainloop()
```

# Tomando datos

```
tk.Label(root, text="Código del artículo:").grid(row=0, column=0, padx=10, pady=5)  
entry_codigo = tk.Entry(root)  
entry_codigo.grid(row=0, column=1)
```

```
tk.Label(root, text="Nombre del artículo:").grid(row=1, column=0, padx=10, pady=5)  
entry_nombre = tk.Entry(root)  
entry_nombre.grid(row=1, column=1)
```

```
tk.Label(root, text="Precio del artículo:").grid(row=2, column=0, padx=10, pady=5)  
entry_precio = tk.Entry(root)  
entry_precio.grid(row=2, column=1)
```

# Evento agregar

# Función para agregar un nuevo artículo

```
def agregar_articulo():
```

```
    codigo = entry_codigo.get()
```

```
    nombre = entry_nombre.get()
```

```
    precio = entry_precio.get()
```

```
    messagebox.showwarning("Ejemplo", codigo + " " + nombre + " " + precio)
```

```
.....
```

```
tk.Button(root, text="Agregar", command=agregar_articulo).grid(row=3, column=0,  
pady=10)
```

# Agregando la lista

```
# Crear la lista de artículos
```

```
lista_articulos = tk.Listbox(root, width=50, height=10)
```

```
lista_articulos.grid(row=5, column=0, columnspan=2, padx=10, pady=10)
```

# Agregando elementos a la lista

```
# Lista global para almacenar los artículos
articulos = []
# Función para agregar un nuevo artículo
def agregar_articulo():
    codigo = entry_codigo.get()
    nombre = entry_nombre.get()
    precio = entry_precio.get()
    # Agregar el artículo a la lista
    articulos.append({"codigo": codigo, "nombre": nombre, "precio": float(precio)})
    actualizar_lista()
    # Limpiar los campos
    entry_codigo.delete(0, tk.END)
    entry_nombre.delete(0, tk.END)
    entry_precio.delete(0, tk.END)
# Función para actualizar la lista en la interfaz
def actualizar_lista():
    lista_articulos.delete(0, tk.END) # Limpiar la lista
    for articulo in articulos:
        lista_articulos.insert(tk.END, f"{articulo['codigo']} - {articulo['nombre']} - {articulo['precio']:.2f}")
```



# Eliminar un artículo

# Función para eliminar un artículo seleccionado

```
def eliminar_articulo():
```

```
    seleccion = lista_articulos.curselection()
```

```
    if seleccion:
```

```
        index = seleccion[0]
```

```
        del articulos[index]
```

```
        actualizar_lista()
```

```
    else:
```

```
        messagebox.showwarning("Advertencia", "Debes seleccionar un artículo para eliminar")
```

```
.....
```

```
tk.Button(root, text="Eliminar", command=eliminar_articulo).grid(row=3, column=1, pady=10)
```

# Seleccionar un artículo

```
# Función para seleccionar un artículo para modificar
def seleccionar_articulo():
    seleccion = lista_articulos.curselection()
    if seleccion:
        index = seleccion[0]
        articulo = articulos[index]

        # Llenar los campos con los datos seleccionados
        entry_codigo.delete(0, tk.END)
        entry_nombre.delete(0, tk.END)
        entry_precio.delete(0, tk.END)

        entry_codigo.insert(tk.END, articulo["codigo"])
        entry_nombre.insert(tk.END, articulo["nombre"])
        entry_precio.insert(tk.END, articulo["precio"])
    .....

tk.Button(root, text="Seleccionar", command=seleccionar_articulo).grid(row=3, column=2, pady=5)
```

# Modificar un artículo

```
# Función para modificar un artículo seleccionado
def modificar_articulo():
    seleccion = lista_articulos.curselection()
    if seleccion:
        index = seleccion[0]
        codigo = entry_codigo.get()
        nombre = entry_nombre.get()
        precio = entry_precio.get()
        # Actualizar el artículo
        articulos[index] = {"codigo": codigo, "nombre": nombre, "precio": float(precio)}
        actualizar_lista()
        # Limpiar los campos
        entry_codigo.delete(0, tk.END)
        entry_nombre.delete(0, tk.END)
        entry_precio.delete(0, tk.END)
    else:
        messagebox.showwarning("Advertencia", "Debes seleccionar un artículo para modificar")
    .....
tk.Button(root, text="Modificar", command=modificar_articulo).grid(row=3, column=4, pady=5)
```