

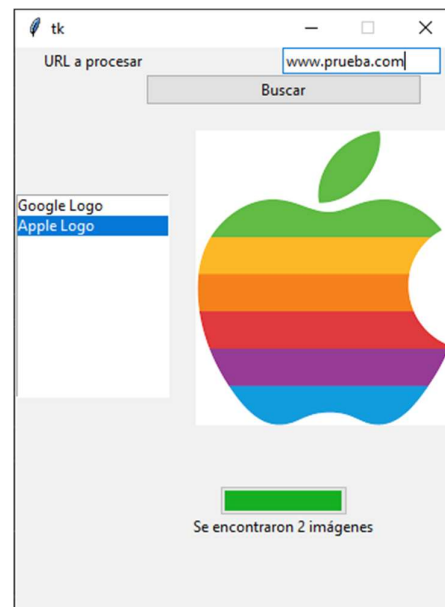
## Paradigmas de Programación: Práctica 2

Vamos a crear una aplicación combinando tres de las herramientas vistas hasta ahora: **ReactiveX (rxpy)**, **Tkinter** y **AsyncIO**.

La aplicación consistirá en una ventana gráfica en la que el usuario puede introducir una URL. Al introducirla y pulsar el botón de buscar, descargaremos la página web que se encuentre en la URL introducida y, de ella, procesaremos todas las imágenes descargando sus datos a memoria (no en disco). La descarga de imágenes debe realizarse de forma concurrente con asyncio utilizando la librería aiohttp y, cada vez que descarguemos una, notificaremos a la ventana principal de la aplicación por medio de un Observable-Observer de ReactiveX, aplicando pipes siempre que sea necesario

La ventana principal mostrará tres elementos más: una lista (Listbox), una barra de progreso (Progressbar) y una imagen (PIL.ImageTk.PhotoImage)

- La barra de progreso se irá rellenando paso a paso cada vez que se complete la descarga de una imagen (el progreso de la barra se puede establecer con el método `step`, con valores de 0 a 99.99). Debajo de la barra mostraremos un texto que diga “Se encontraron N imágenes”, mostrando el total de *imgs* detectadas en la web
- La lista tendrá un elemento nuevo por cada imagen que descarguemos, que mostrará el nombre de la imagen (podemos sacarlo del atributo “alt” de la `img` del html o del propio nombre de la imagen original). **Es importante que no escribamos en disco los datos de la imagen**, almacenaremos el conjunto de bytes en memoria.
  - Por último, si seleccionamos una imagen del listado, la mostraremos a la derecha, como se ve en el ejemplo.



Las librerías a utilizar, como mínimo, serán: `TKInter`, `Pillow`, `rx`, `aiohttp`, `asyncio`, y podeis utilizar `beautifulsoup4` para parsear el html

La práctica obtendrá una puntuación de 0 a 9, evaluando el funcionamiento correcto **sin interbloqueos**, el correcto uso de las arquitecturas especificadas (programación reactiva y asíncrona), y el funcionamiento de la UI

- El punto restante para alcanzar el 10 se obtendrá si se implementa la aplicación sin interbloqueos evitando usar `Threads`, solo con `asyncio`.

El no uso, o la utilización incorrecta de alguna de las arquitecturas (arquitectura reactiva y arquitectura asíncrona con corrutinas) supondrá que se evalúe esta práctica como suspenso