

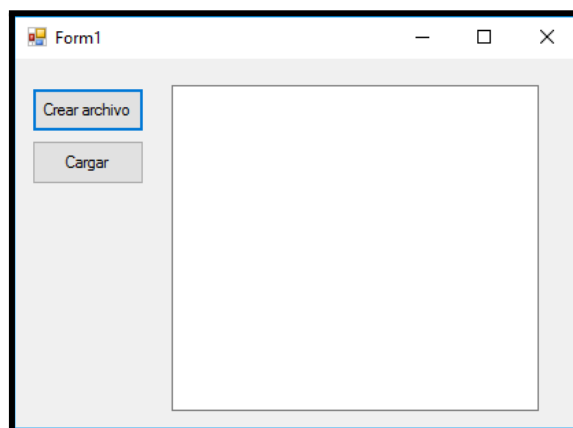
## LINQ to XML:

- Tareas que hacemos usualmente los desarrolladores cómo cargar documentos XML en memoria, realizar consultas y hasta modificar, resultan más sencillas
- Otras funciones importantes que nos trae esta API:
  - podemos guardar documentos XML existentes en memoria y serializarlos;
  - evitamos la tarea de aprender el lenguaje de consulta XML

### Objetivo de la tarea:

- *Crear árboles XML: aprender a crear elementos XML y documentos XML mediante el uso de “literales XML”*
- *Serializar un objeto XDocument en un archivo. Serialización: proceso de codificación de un objeto en un medio de almacenamiento con el fin de transmitirlo a través de una conexión en red como una serie de bytes o en un formato humanamente más legible como XML o JSON, entre otros.*
- *Aprender a cargar XML desde un archivo, cadena o secuencia*

**Tarea:** Aplicación que haciendo uso de Linq to Xml genere el archivo **Contactos.xml** y muestre su contenido en un textBox



- ★ **Botón “Crear archivo”.** Su pulsación debe crear en disco el archivo **Contactos.xml** a partir de literales xml. Contenido del archivo a crear:

```
<?xml version="1.0" encoding="utf-8" ?>
<Contactos>
  <Contacto>
    <Nombre>Juan</Nombre>
    <Email>juan@gmail.com</Email>
    <Telefono Móvil="666666666" Trabajo="911111111" />
  </Contacto>
  <Contacto>
    <Nombre>Luis</Nombre>
    <Email>luis@gmail.com</Email>
    <Telefono Móvil="666777777" />
  </Contacto>
  <Contacto>
    <Nombre>María</Nombre>
    <Email>maria@gmail.com</Email>
    <Telefono Móvil="666888888" Personal="91222222" />
  </Contacto>
  <Contacto>
    <Nombre>Lola</Nombre>
```



## Tarea1-Crear y serializar un XDocument en un archivo

DAM2

```
<Email>lola@gmail.com</Email>  
<Telefono Móvil="666888888" Personal="91222222" />  
</Contacto>  
</Contactos>
```

- ★ **Botón “Cargar”**. Muestra en un TextBox multiline el contenido del archivo anteriormente creado. (*método **load***)

*Nota:* la siguiente sentencia muestra doc en el textbox multiline txtRes, siendo doc un XDocument

```
txtRes.Text = doc.ToString()
```