
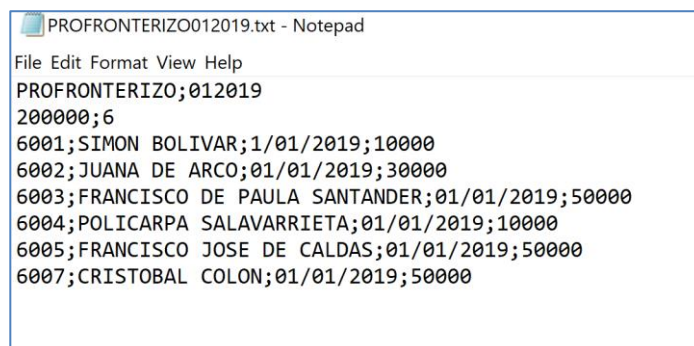
 <b>UNIVERSIDAD</b> Popular del Cesar	<b>UNIVERSIDAD POPULAR DEL CESAR</b> <b>PROGRAMACIÓN III</b> PARCIAL 2	<b>2019-II</b>
---	--	----------------

Nombre Completo: \_\_\_\_\_

La Alcaldía de Valledupar requiere una aplicación para realizar el reporte de las liquidaciones del impuesto de Estampilla que debe ser pagado por sus contratistas a la Gobernación del Cesar, las estampillas que recauda la Alcaldía para la Gobernación son: Estampilla de Pro-Adulto Mayor, Estampilla ProDesarrollo Fronterizo y Estampilla ProDesarrollo Departamental.

- Con este fin, la Alcaldía requiere registrar las liquidaciones pagadas por sus contratistas del impuesto de estampilla para lo cual solicita: Identificación del Contratista, Nombre del contratista, Impuesto de estampilla, fecha en que se liquidó el impuesto y valor del impuesto.
- La Alcaldía también requiere consultar las liquidaciones, esta consulta, debe visualizar las liquidaciones por cada tipo de estampilla en un mes de un año específico y el valor total de esas estampillas. Por tal motivo, la consulta debe permitir el filtro por Mes y año y por tipo de estampilla, además de visualizar el valor total de las liquidaciones según el filtro realizado.
- Finalmente, la Alcaldía debe generar un nuevo archivo plano con las liquidaciones visualizadas en la consulta anterior, como se muestra en la **Figura 1**. Para esto tenga en cuenta que:
  - a) El nombre del nuevo archivo se deberá generar con el nombre: Estampilla+ el mes del reporte + el año en que se genera el reporte.  
Por ejemplo: si se genera el reporte de la estampilla de Profonterizo del mes de enero del 2019, el nombre del archivo será el siguiente.  
 PROFRONTERIZO012019.txt - Notepad
  - b) El reporte tendrá el siguiente formato.
    - En la primera línea del archivo, visualizar el nombre de la Estampilla a reportar, el mes y año del reporte.
    - En la segunda línea del archivo, visualizar el valor total de la liquidación de la estampilla seleccionada y la cantidad de liquidaciones reportadas.
    - En las líneas siguientes, visualizar cada una de las liquidaciones a los contratistas de la Alcaldía de acuerdo al filtro.




```

PROFRONTERIZO012019.txt - Notepad
File Edit Format View Help
PROFRONTERIZO;012019
200000;6
6001;SIMON BOLIVAR;1/01/2019;10000
6002;JUANA DE ARCO;01/01/2019;30000
6003;FRANCISCO DE PAULA SANTANDER;01/01/2019;50000
6004;POLICARPA SALAVARRIETA;01/01/2019;10000
6005;FRANCISCO JOSE DE CALDAS;01/01/2019;50000
6007;CRISTOBAL COLON;01/01/2019;50000

```

Figura 1. Ejemplo del Archivo a exportar

 <b>UNIVERSIDAD</b> Popular del Cesar	UNIVERSIDAD POPULAR DEL CESAR <b>PROGRAMACIÓN III</b> PARCIAL 2	<b>2019-II</b>
---	---	----------------

Nombre Completo: \_\_\_\_\_

A partir de los requerimientos anteriores, realizar una aplicación con arquitectura en capas, manejo de repositorio en archivo y consultas a través de expresiones Lambda para:

1. (1.25) Crear la Interfaz gráfica de usuario y demás clases que permita registrar las liquidaciones
2. (1.25) Crear la interfaz gráfica de usuario y demás clases que permita consultar las liquidaciones de acuerdo al filtro por Impuesto, mes y año.
3. (2.5) Generar el reporte en un nuevo archivo plano de las liquidaciones de acuerdo al filtro por Impuesto, mes y año teniendo en cuenta el formato establecido en la **Figura 1**.

**Nota:**


- El atributo fecha del servicio debe ser **DateTime**.
- Para solicitar la fecha desde la interfaz de usuario utilizar la herramienta **DateTimePicker**.
- Para obtener el valor de la fecha de un **DateTimePicker** utilice la propiedad value ejemplo **DtpFecha.Value.Date**
- Para guardar la fecha en el archivo con el formato requerido puede utilizar **FechaServicio.ToShortDateString()** o **FechaPago.ToString("dd/MM/yyyy")**.
- Para los filtros para extraer el mes de un DateTime utilice **fecha.Month** y para extraer el año de un DateTime utilice **fecha.Year**
- Para nombrar del reporte RIPS utilice la forma anterior o **FechaPago.ToString("MMyyyy")**.

Rutinas Guía para Leer del Archivo y Escribir en el Archivo. (**Ojo Nombrar las clases de acuerdo al Dominio del Problema del parcial esto es solo una Guía, no la solución del parcial.**)

```
public void Guardar(Recaudo recaudo)
{
    FileStream origen = new FileStream(ruta, FileMode.Append);
    StreamWriter writer = new StreamWriter(origen);
    writer.WriteLine(recaudo.Entidad + ";" + recaudo.NumeroRecibo + ";" +
recaudo.FechaPago.ToString("dd/MM/yyyy") + ";" + recaudo.ValorPago);
    writer.Close();
    origen.Close();
}
```

```
public List<Recaudo> Consultar()
{
    List<Recaudo> recaudos = new List<Recaudo>();
    FileStream origen = new FileStream(ruta, FileMode.OpenOrCreate);
    StreamReader reader = new StreamReader(origen);
    string linea=string.Empty;
    while ((linea = reader.ReadLine()) != null)
    {
        Recaudo recaudo = Mapear(linea);
        recaudos.Add(recaudo);
    }
    reader.Close();
    origen.Close();
    return recaudos;
}

private Recaudo Mapear(string linea)
```

 <b>UNIVERSIDAD</b> Popular del Cesar	<b>UNIVERSIDAD POPULAR DEL CESAR</b> <b>PROGRAMACIÓN III</b> <b>PARCIAL 2</b>	<b>2019-II</b>
---	---	----------------

Nombre Completo: \_\_\_\_\_

```

{
    string[] datos = linea.Split(';');
    Recaudo recaudo = new Recaudo();
    recaudo.Entidad = datos[0];
    recaudo.NumeroRecibo = datos[1];
    recaudo.FechaPago = DateTime.Parse(datos[2]);
    recaudo.ValorPago = decimal.Parse(datos[3]);
    return recaudo;
}

```