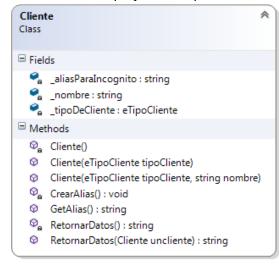
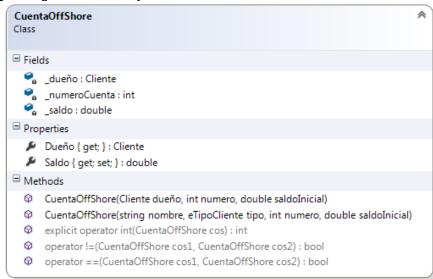
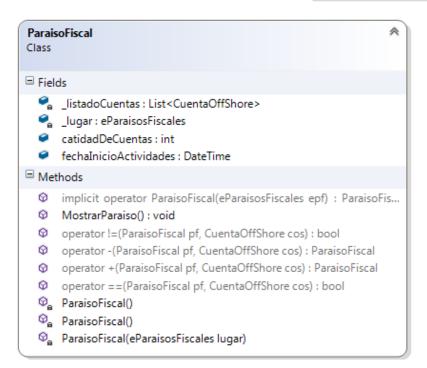
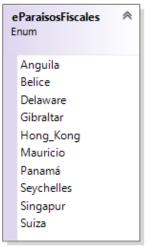
# LABORATORIO II - 2020

Desarrollar en un proyecto de tipo Class Library las siguientes clases y enumerados.











### Clase Cliente

- 1. **Constructor**: El constructor (privado) por default inicializará al nombre con "NN", al alias con "Sin alias" y al tipo como eTipoCliente.SinTipo. Las otras sobrecargas deberán inicializarán los atributos que correspondan, siempre reutilizando las demás sobrecargas.
- 2. Métodos de instancia:
  - El método privado "CrearAlias" inicializará al atributo \_aliasParaIncognito con un número al azar (utilizar Random de 1000 a 9999) concatenado al atributo \_tipoDeCliente. Ejemplo: "1536Financista".
  - El método público "GetAlias" se encargará de retornar el alias del cliente. Si el valor del atributo aliasParaIncognito es igual a "Sin alias", se deberá invocar al método descrito anteriormente.
  - El método privado "RetornarDatos" retornará la información del Cliente. Usar la clase **StringBuilder**.
  - El método público y de clase "RetornarDatos" retornará un string conteniendo los datos de la instancia de tipo Cliente que recibe como parámetro. Reutilizar código.

#### Clase CuentaOffShore

- 3. Constructor: Reutilizar código entre las sobrecargas.
- 4. Sobrecarga de operadores:
  - **Igualdad** (==): Retornará un true, si ambas CuentasOffShore poseen el mismo número de cuenta y pertenecen al mismo dueño (utilizar GetAlias), false caso contrario. Reutilizar código.
  - Explícito: Retornará el número de cuenta de la CuentaOffShore. Ejemplo int nro = (int)objCoffShore;

### Clase ParaisoFiscal

- 5. Constructor: (todas las sobrecargas privadas): El constructor de clase será el encargado de inicializar los atributos estáticos cantidadDeCuentas (con cero) y fechalnicioActividades (con DateTime.Now).
  El resto de las sobrecargas son del constructor de instancia. El constructor (por default) será el único encargado de inicializar la lista de CuentasOffShore. Reutilizar código.
- 6. Método de instancia:
  - MostrarParaiso (público): Muestra toda la información disponible para cada paraíso fiscal (fecha de inicio de actividades, lugar de radicación, cantidad de cuentas offshore que posee y el detalle de cada una de ellas).
     Reutilizar código.

## 7. Sobrecarga de operadores:

- Implícito: Recibe un eParaisosFiscales y retorna un objeto de tipo ParaisoFiscal.
   Ejemplo: ParaisoFiscal pf = eParaisosFiscales.Panamá;
- Igualdad (==): Retornará un true, si la CuentaOffShore existe en el ParaisoFiscal, false caso contrario.
   Reutilizar código.
- Sustracción (-): Permite quitar una CuentaOffShore a la lista de cuentas de la clase si la misma se encuentra en el ParaisoFiscal. Decrementar en uno el atributo cantidadDeCuentas si se pudo quitar la cuenta. Reutilizar código. Informar de lo acontecido.
- Adición (+): Permite agregar una CuentaOffShore a la lista de cuentas de la clase, sólo si la misma no se encuentra en el ParaisoFiscal. Incrementar en uno el atributo cantidadDeCuentas.
   Si la CuentaOffShore existe en el ParaisoFiscal, se debe sumar los saldos.
   Reutilizar código. Informar de lo acontecido.

## Este es el Main:

```
Console.Title = "Apellido y Nombre";//Cambiar por SU apellido y
SU nombre
Cliente mauri = new Cliente(eTipoCliente.PoliticoCorrupto,
Cliente fariña = new Cliente(eTipoCliente.Financista, "Fariña");
Cliente mesias = new Cliente(eTipoCliente.JugadorDeFutbol,
"Lio");
CuentaOffShore messiOff = new CuentaOffShore(mesias, 123, 15000);
CuentaOffShore mauriOff = new CuentaOffShore(mauri, 678, 25000);
CuentaOffShore lazaroOff = new CuentaOffShore("Lázaro",
eTipoCliente.EmpresarioCorructo, 456, 56000);
CuentaOffShore otraDeMauri = new CuentaOffShore(mauri, 678,
CuentaOffShore fariOff = new CuentaOffShore(fariña, 666, 3500);
ParaisoFiscal panamaPappers = eParaisosFiscales.Panamá;
panamaPappers += messiOff;
panamaPappers += mauriOff;
panamaPappers += lazaroOff;
panamaPappers.MostrarParaiso();
panamaPappers += otraDeMauri;
panamaPappers -= messiOff;
panamaPappers -= fariOff;
panamaPappers.MostrarParaiso();
Console.ReadLine();
```

## Salida de Consola: