

EVALUACION	Examen – Cursos dictados desde Marzo 2019 en adelante	GRUPO	TODOS	FECHA	30/10/2020
MATERIA	PROGRAMACIÓN 2				
CARRERA	Analista Programador / Analista en Tecnologías de la Información				
CONDICIONES	<ul style="list-style-type: none"> - Puntos: 100 - Duración: 3 Hrs - Sin material - No escriba la hoja de la letra - Consultas solamente sobre interpretación de la letra y sintáxis específica del lenguaje. 				

Ejercicio 1

VigilantORT es una empresa de vigilancia creada por egresados de la Universidad ORT, quienes vieron un nicho en el mercado en el que la delincuencia en el Uruguay ha crecido y no hay tantas empresas que provean dichos servicios. VigilantORT provee paquetes de servicios para sus distintos clientes, los cuales son instalados por distintos instaladores.

De cada instalador se conoce su nombre (único), su teléfono y la fecha de ingreso en la empresa. Sobre los servicios que provee la empresa se conoce el nombre de éste, la duración (en meses) y un costo base, el cual es el mismo para todos los servicios. De cada cliente se conoce un id autogenerado, su nombre, dirección y el instalador asignado por la empresa.

Los clientes pueden realizar distintas contrataciones de servicios, de las cuales sólo interesa saber la fecha de contratación. Un mismo servicio puede ser contratado por varios clientes, y un mismo cliente puede contratar varios servicios.

Existen dos tipos distintos de servicios:

- Servicio Básico: Se conoce la cantidad de sensores de movimiento incluidos en el contrato y si posee respuesta a disparos de alarma o no.
- Servicio Full: Se conoce una prima o costo extra (particular para cada Servicio Full) y la cantidad de cámaras contratadas.

El cálculo del costo del servicio depende de su tipo. Todos los servicios cuestan el monto base más el siguiente cálculo:

- Servicio Básico: \$200 por cada sensor. Si el servicio tiene respuesta a disparos de alarma cuesta \$189.1 más
- Servicio Full: prima más un 2% por cada cámara.

Se pide:

- a) Diagrama de clases completo que permita resolver los siguientes requerimientos: **20 puntos**

R01- Dado el id de un cliente, obtener el costo de todos los servicios contratados por el mismo.

R02- Devolver el o los instaladores asignados a la mayor cantidad de clientes.

R03- Devolver una lista con todas las contrataciones realizadas en el último año calendario (365 días) ordenadas por fecha descendientemente.

- b) Implementar solo a nivel de dominio los requerimientos R01 (**20 puntos**), R02 (**15 puntos**) y R03 (**15 puntos**)

Importante: En el requerimiento R01 es obligatorio usar polimorfismo para su implementación, en caso de no usarlo no se obtendrán puntos por la resolución de ese requerimiento.

Ejercicio 2 (20 puntos)

Tomando en cuenta el diagrama de clases definido en el ejercicio 1 parte a), implementar lo que considere necesario a nivel de la vista y el controlador para listar todos los clientes registrados en el sistema, indicando la cantidad de contrataciones que tiene cada uno.

Ejercicio 3 (10 puntos)

Indique cuál o cuáles de las siguientes afirmaciones son verdaderas o falsas, justificando la respuesta en todos los casos:

- i. Si trabajo con el modelo a nivel de MVC, y tengo la siguiente estructura en la clase que utilizo como modelo y en la vista:

```
// Modelo
public class Mascota

    private int numeroChip;
    private string nombre;
    private string raza;

    public Mascota(int numeroChip, string nombre, string raza){
        this.numeroChip = numeroChip;
        this.nombre = nombre;
        this.raza = raza;
    }
}

//Vista
@model Proyecto.Mascota

<form method="post" action="~/Mascota/Alta">
<label>Ingrese número de chip</label>
<input type="number" value=@Model.NumeroChip name="numeroChip">
<label>Ingrese nombre de la mascota</label>
<input type="text" value=@Model.Nombre name="nombre">
<label>Ingrese raza de la mascota</label>
<input type="text" value=@Model.Raza name="raza">
<input type="button" value="Guardar">
</form>
```

- a) La clase que utilizo como modelo, posee la estructura correcta

-
- b) La implementación no es correcta, para que funcione es necesario agregar en la clase Mascota las properties referidas dentro del form.
 - c) La implementación no es correcta, ya que al apretar el botón no sucedería nada.
 - d) La implementación no es correcta, es necesario agregar en la clase Mascota el constructor con parámetros
 - e) La implementación realizada a nivel de la vista y la clase Mascota es correcta.
- ii. El ViewBag es utilizado para almacenar datos.
- a) Se utiliza para pasar datos del controlador a la vista y viceversa
 - b) Puedo almacenar cualquier tipo de elementos.
 - c) Una vez que guardo un elemento de un tipo X, todos los demás elementos deberán ser también de tipo X.
 - d) Puede ser usado tanto en la aplicación web como en el dominio.
- iii. Se tiene la siguiente URL: localhost.1891/Mascota/BuscarPorChip?chip=1891
- a) Se tiene que haber implementado un controlador Mascota con el método BuscarPorChip
 - b) Se tiene que haber implementado un controlador Mascota con el método Index
 - c) El parámetro chip llega al controlador dentro del ViewBag
 - d) La acción a la que se llama debe estar explícitamente marcada con el atributo HttpGet.