

EVALUACION	Examen Cursos dictados desde Marzo 2019 en adelante	FECHA	06/08/2019
MATERIA	PROGRAMACIÓN 2		
CARRERA	Analista Programador / Analista en Tecnologías de la Información		
CONDICIONES	- 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

RepartORT es una empresa nacional formada por exalumnos de la Universidad ORT, creada con el fin de darle trabajo a miles de personas. La empresa es Eco-Friendly, ya que sólo trabaja con repartidores en bicicleta, obligándoles a usar casco, chaleco reflectivo y a poseer el carné de manipulador de alimentos.

RepartORT posee varios empleados, de los cuales se conoce su cédula (única), fecha de nacimiento, nombre completo y grupo sanguíneo (A, B, AB, O). Existen dos tipos de empleados:

- **Jornalero:** Se conoce la cantidad de horas semanales (entre 3 y 8), el sueldo por hora y si realiza trabajo nocturno.
- **Mensual:** Se conoce la fecha de contratación y el sueldo mensual.

Cada empleado tiene un sueldo base, el cual es mensual y es común para ambos tipos de empleados.

Para calcular el sueldo de un empleado, se toma como mínimo el sueldo base de los empleados. Si el empleado es jornalero, se le suma el cuádruple de las horas semanales por el sueldo por hora. Si trabaja en horario nocturno, se le suma un 10% extra. Si el empleado es mensual, se le suma el sueldo mensual al sueldo base. Además, por cada seis meses trabajados, se le suma un 1,2% de sueldo en concepto de antigüedad.

Las bicicletas son de la empresa. De cada bicicleta se conoce marca, año, rodado y un código (autonumerado). Cada empleado es asignado a una bicicleta, e interesa saber de cada bicicleta qué empleados la utilizaron históricamente.

Los repartidores son asignados a un barrio cada uno. De cada barrio se debe conocer el nombre y el costo de envío a domicilio.

Se pide:

1) Realizar el diagrama de clases del dominio en UML que modele la realidad planteada y permita resolver las siguientes operaciones (35 puntos):

- Alta de empleado
- Alta de bicicleta
- Dada la cédula de un empleado calcular el sueldo en el último mes.
- Dado un código de bicicleta indicar si la misma fue utilizada históricamente por más de 10 empleados.

2) Implemente en C# .NET (solo métodos del dominio) las operaciones c (20 puntos) y d (15 puntos) de la parte anterior.

- 3) Dado el modelo MVC, escribir el código correspondiente a la View y al Controller necesario para codificar el alta de bicicleta. En caso de error o de éxito, se deberá mostrar en la misma pantalla de ingreso un mensaje informativo del resultado de la operación. La pantalla deberá tener un diseño similar al siguiente. Se asume que se cuenta con la implementación del método del dominio para dar de alta a la bicicleta indicado en la parte 1.a (15 puntos):

Marca

Año

Rodado

Mensaje con el resultado de la operación

Notas:

- Se valorará especialmente la eficiencia de los algoritmos implementados.
- No es necesario representar en el diagrama las propiedades ni los constructores, los cuales se asumen dados.
- El diagrama deberá incluir las relaciones entre clases (con su cardinalidad, navegabilidad, tipo de relación y los adornos que sean necesarios), los atributos con sus tipos de datos y las firmas de los métodos (principales y accesorios) con su visibilidad, lista de parámetros y retornos.
- El diseño deberá asegurar que se cree un único objeto de la clase administradora.
- Se valorará especialmente la buena delegación de responsabilidades.
- Deberá implementar todo método auxiliar que utilice.

Ejercicio 2 (15 puntos)

- i. Justificando en TODOS los casos, indique Verdadero o Falso para las siguientes afirmaciones:
1. Dada la siguiente URL: <http://localhost:1891/usuario/login>
 - a. La aplicación se está ejecutando en el IIS Local de la máquina
 - b. Existe un LoginController en este caso
 - c. Existe un UsuarioController, el cual contiene una acción Login.
 2. Si trabajo a nivel de la vista con el modelo...
 - a. no es necesario que la clase que representa el modelo, tenga un constructor vacío
 - b. necesito indicar la etiqueta @model a nivel de la vista
 - c. necesito que la clase que representa el modelo, tenga un constructor con parámetros
 3. Sobre el ViewBag...
 - a. ViewBag puede ser accedido tanto en el código de un ActionResult como en una vista
 - b. ViewBag puede guardar todo tipo de objetos

-
- c. Si cargo un nuevo valor a la ViewBag, éste puede ser consumido en todos los próximos requests

ii. Responda las siguientes consultas:

1. Si en lugar de utilizar el modelo, utilizó parámetros para mapear los datos desde una vista a un controlador. Indique cuál o cuáles opciones tengo que tener en cuenta:
 - a. tengo que indicar la etiqueta @model en la vista
 - b. el name de los controles html tiene que coincidir con el nombre de los parámetros de la action en el controlador
 - c. el name de los controles html tiene que coincidir con el nombre de la property en el modelo
2. Si tengo la siguiente vista llamada ListadoCursos, a la cual se accede desde una opción de menú que tiene indicada la siguiente ruta: `http://localhost:XXXX/Curso/Index`

```
@model List<Curso>
{
    @title ="Mi pagina"
}
<h2>Mi pagina de prueba</h2>
@foreach(Curso c in @model){
    <p>@c.Nombre</p>
    <p>@c.Duracion</p>
}
```

Marque las opciones que no son correctas, para que esta vista funcione en forma correcta

- a. Necesito que exista un controlador llamado Curso, que tenga un action HttpPost
- b. Necesito que exista un controlador llamado Curso, que tenga un action Index
- c. La action Index del controlador, debe tener el siguiente código:

```
public ActionResult Index(){
    return View();
}
```