

Proyecto Programación Orientada a Objetos 2020

Proyecto N°2

Caso:

Una empresa de videojuegos necesita llevar un control de sus trabajadores (desarrolladores, diseñadores y publicistas), ya que actualmente todo el control lo realizan de forma manual, y necesitan de una aplicación que les ayude a agilizar procesos como buscar de forma rápida a un trabajador, crear un trabajador, aumentar sueldo, cambiar el salario, ver la actividad que está haciendo el trabajador, son las funciones que debe tener el sistema.

A continuación, se adjuntan detalles a tener en cuenta para su desarrollo:

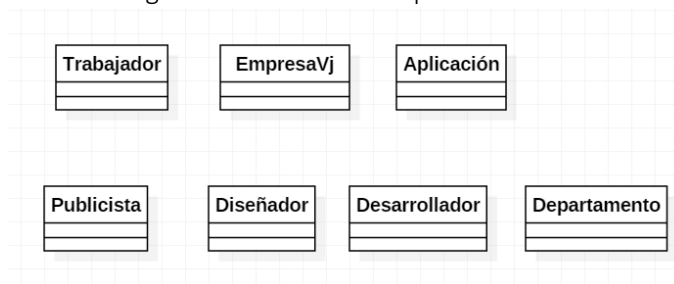
- La empresa, tiene un nombre, una dirección y una lista de trabajadores.
- El trabajador tiene un rut, un nombre, un sueldo y el nombre del juego en el que trabaja.
- Un departamento tiene un nombre, numero de oficina y una descripción

Actividades de los trabajadores.

- Un desarrollador puede estar programando las funcionalidades del videojuego o programando objetos.
- Un diseñador puede estar creando los sprite del juego.
- Un publicista puede estar preparándose para la fecha de lanzamiento del videojuego.

Actividades a desarrollar para el proyecto:

Actividad 1: Crear el Diagrama de clases correspondiente basándote en las siguientes clases:



Actividad 2: Transforme el modelo creado al código Java.

Actividad 3: En la clase empresa, en su constructor, se pide crear la lista de trabajadores y agregar a 5 trabajadores y 2 departamentos, (Producción y otro llamado publicidad y Marketing).

Actividad 4: Se debe buscar a un trabajador, para ello, se debe recorrer la lista, retornando todos los datos del trabajador o NULL en caso de que el trabajador no exista.

Actividad 5: Se debe agregar un nuevo trabajador a la lista, validando que el RUT no se repita, además debe validar que el departamento sea uno de los dos especificados. Debe retornar un String indicando la situación de ingreso correcto o si hubo un problema al ingresar.

Actividad 6: Se debe aumentar el sueldo de un trabajador, para ello se debe buscar al trabajador y aumentar su salario, considerando que lo máximo que se puede aumentar corresponde al 35% del sueldo actual. Debe retornar un String indicando la situación.

Actividad 7: Se debe cambiar el salario, para ello, debe buscar a un trabajador en la lista de trabajadores, si lo encuentra debe cambiar el salario, siempre y cuando el nuevo salario se encuentre en el rango de 200.000 y 500.000 pesos

Actividad 8: Se debe poder ver a todos los trabajadores contratados.

Actividad 9: Se debe poder ver la actividad que está realizando un trabajador en particular según su cargo.

Actividad 10: Se debe crear un menú con todas las funciones operativas.

//////////MENU OPERACIONES//////////

1. CREAR TRABAJADOR
2. BUSCAR TRABAJADOR
3. AUMENTAR SUELDO TRABAJADOR
4. CAMBIAR SUELDO TRABAJADOR
5. VER TRABAJADORES
6. VER LA ACTIVIDAD DE UN TRABAJADOR EN PARTICULAR
7. SALIR

Pauta de Trabajo

Este proyecto debe ser realizado **en parejas**.

- El proyecto debe cumplir con los requerimientos mínimos especificados anteriormente.
- Proyectos que sean considerados copias obtendrán la calificación 1.0.
- La entrega corresponde al **día lunes 30 de noviembre**, a las 23:55 horas. Por el recurso que se encontrará habilitado en Educandus debe incluir el archivo con todas las clases de su proyecto, ("Carpeta proyecto" o "carpeta src" comprimida) y el diagrama de clases.
- No es necesaria una interfaz gráfica.

		NIVELES Y PUNTAJES				
INDICADORES		DESTACADO 7PTS	COMPETENTE 5 PTS	EN DESARROLLO 4 PTS	BÁSICO 3 PTs	POR MEJORAR 1 PT
Construye el diagrama de clases, Establece las relaciones y cardinalidad entre clases	Activ. 1	Construye correctamente el diagrama de clases correspondiente al caso presentado, utilizando atributos con el modificador de acceso – o #	Construye el diagrama de clases correspondiente al caso presentado, pero con errores simples de nomenclatura	Construye el diagrama de clases correspondiente al caso presentado faltando algunas relaciones y/o métodos	Construye el diagrama de clases correspondiente al caso presentado, pero con errores de lógica o nomenclatura.	No realiza la actividad
Declara la estructura de una clase en Java, Establece las relaciones y cardinalidad entre clases, mediante código Java	Activ. 2	Codifica correctamente el modelo expuesto a código Java	Codifica el modelo expuesto a código Java con algunos errores de cardinalidad	Codifica el modelo, faltando algunas relaciones y/o métodos	Codifica el modelo, con errores de programación	No realiza la actividad
Crea constructores para la implementación de la clase	Activ. 3	Crea correctamente los constructores del modelo iniciando con valores respectivos	Crea correctamente los constructores del modelo iniciando con valores respectivos, pero no todas las clases lo encuentran implementado	Crea correctamente cada constructor, pero no hay evidencia de inicio de valores.	Crea los constructores presentando parámetros incorrectos y/o con errores en el código	No realiza la actividad
Implementa métodos correspondientes para una clase, considerando su tipo y retorno	Activ. 4	Implementa correctamente el algoritmo para buscar a un trabajador	Implementa correctamente el algoritmo para buscar a un trabajador, pero lo muestra de manera errónea por pantalla	El método cumple el rango de cumplimiento 70% a un 99%	El método cumple el rango de cumplimiento inferior a 70% O no trabaja con las clases involucradas en la solución.	No realiza la actividad.

Implementa métodos correspondientes para una clase, considerando su tipo y retorno	Actv. 5	Implementa correctamente el ingreso del trabajador en base al requerimiento, respetando que no existe un límite de ingreso de trabajadores	Implementa correctamente el ingreso del trabajador en base al requerimiento, e indica la situación del ingreso.	El método cumple el rango de cumplimiento 70% a un 99%	El método cumple el rango de cumplimiento inferior a 70% O no trabaja con las clases involucradas en la solución.	No realiza la actividad
Implementa métodos correspondientes para una clase, considerando su tipo y retorno	Actv 6	Implementa correctamente el aumento de salario del trabajador en base al requerimiento	Implementa correctamente el aumento de salario del trabajador en base al requerimiento e indica la situación del aumento.	El método cumple el rango de cumplimiento 70% a un 99%	El método cumple el rango de cumplimiento inferior a 70% O no trabaja con las clases involucradas en la solución.	No realiza la actividad
Implementa métodos correspondientes para una clase, considerando su tipo y retorno	Actv 7	Implementa correctamente el cambio de salario de un trabajador en base al requerimiento	Implementa correctamente el cambio de salario de un trabajador en base al requerimiento, pero no respeta los límites indicados	El método cumple el rango de cumplimiento 70% a un 99%	El método cumple el rango de cumplimiento inferior a 70% O no trabaja con las clases involucradas en la solución.	No realiza la actividad
Implementa métodos correspondientes para una clase, considerando su tipo y retorno	Actv 8	Implementa correctamente el retorno de los trabajadores contratados en base al requerimiento	Implementa correctamente el retorno de los trabajadores contratados en base al requerimiento, pero no se observan claramente por pantalla.	El método cumple el rango de cumplimiento 70% a un 99%	El método cumple el rango de cumplimiento inferior a 70%. O no trabaja con las clases involucradas en la solución.	No realiza la actividad
Implementa métodos correspondientes para una clase, considerando su tipo y retorno	Actv 9	Implementa correctamente el retorno de la actividad	Implementa correctamente el retorno de la actividad	El método cumple el rango de cumplimiento 70% a un 99%	El método cumple el rango de cumplimiento inferior a 70%	No realiza la actividad

		realizada por el trabajador en base al requerimiento	realizada por el trabajador en base al requerimiento, pero no se muestra claramente por pantalla.			
Organiza la información a presentar por la actividad	Actv 10	Implementa correctamente las llamadas a las funciones en base al requerimiento	Implementa las llamadas a las funciones en base al requerimiento, pero existen errores en su implementación.	El método cumple el rango de cumplimiento 70% a un 99%	El método cumple el rango de cumplimiento inferior a 70% O no trabaja con las clases involucradas en la solución.	No realiza la actividad

Escala de Notas

10 puntos : 1.00	26 puntos : 2.33	41 puntos : 3.58	56 puntos : 5.25
11 puntos : 1.08	27 puntos : 2.42	42 puntos : 3.67	57 puntos : 5.38
12 puntos : 1.17	28 puntos : 2.50	43 puntos : 3.75	58 puntos : 5.50
13 puntos : 1.25	29 puntos : 2.58	44 puntos : 3.83	59 puntos : 5.62
14 puntos : 1.33	30 puntos : 2.67	45 puntos : 3.92	60 puntos : 5.75
15 puntos : 1.42	31 puntos : 2.75	46 puntos : 4.00	61 puntos : 5.88
16 puntos : 1.50	32 puntos : 2.83	47 puntos : 4.12	62 puntos : 6.00
17 puntos : 1.58	33 puntos : 2.92	48 puntos : 4.25	63 puntos : 6.12
18 puntos : 1.67	34 puntos : 3.00	49 puntos : 4.38	64 puntos : 6.25
19 puntos : 1.75	35 puntos : 3.08	50 puntos : 4.50	65 puntos : 6.38
20 puntos : 1.83	36 puntos : 3.17	51 puntos : 4.62	66 puntos : 6.50
21 puntos : 1.92	37 puntos : 3.25	52 puntos : 4.75	67 puntos : 6.62
22 puntos : 2.00	38 puntos : 3.33	53 puntos : 4.88	68 puntos : 6.75
23 puntos : 2.08	39 puntos : 3.42	54 puntos : 5.00	69 puntos : 6.88
24 puntos : 2.17	40 puntos : 3.50	55 puntos : 5.12	70 puntos : 7.00
25 puntos : 2.25			