

## Algorítmica y Programación orientada por Objetos 1

### Ejercicio Nivel 4

### Parque Empresarial

#### Descripción Global

Una empresa constructora de edificios necesita un programa que le permita administrar la construcción de un parque empresarial. Un parque empresarial es un área donde se construyen edificios de carácter únicamente comercial, no industrial o residencial; con el fin de garantizar la calidad laboral de los trabajadores.

#### Objetivos del ejercicio

El objetivo de este ejercicio es que el estudiante comprenda y adquiera práctica en:

- Asignación de responsabilidades a clases.
- Definición y uso de contratos de métodos.
- Uso de la herramienta de documentación Javadoc.
- Uso de excepciones (clase `Exception`) para indicar situaciones de error en la aplicación.

#### Preparación

Esta sección presenta una lista de chequeo de todas las tareas necesarias para la preparación del ejercicio. Por favor, revise que cada tarea haya sido completada **antes** de pasar a la siguiente sección de esta guía de trabajo.

1. Para conocer el funcionamiento esperado de la aplicación, descargue y/o ejecute el archivo demo del ejercicio que se encuentra en el siguiente enlace: <https://cupi2.virtual.uniandes.edu.co/ejercicios-del-semester-apo1/ejercicio-n4>
2. Descargue el esqueleto del ejercicio (<https://cupi2.virtual.uniandes.edu.co/ejercicios-del-semester-apo1/ejercicio-n4>).
3. Descomprima este archivo e importe el proyecto llamado **n4\_parqueEmpresarial** a Eclipse.

El siguiente video explica detalladamente la forma en la cual se descarga, descomprime e importa a Eclipse, para Windows (en el enlace: <https://youtu.be/4PDAn55uNTI>) y para MAC (en el siguiente enlace: <https://youtu.be/XlkMktjIN6c>). Es muy importante que consulte este video, porque lo explicado allí se realizará a lo largo de todo el semestre en

el curso y usted deberá hacerlo con especial agilidad en todos los laboratorios y parciales prácticos.

Finalmente, limpie el ejercicio, para ello en Eclipse vaya a: **Project > Clean > Ok**, en el siguiente enlace hay un video con un ejemplo de cómo limpiar un proyecto en Eclipse: <https://youtu.be/mbcpY46wXS0>.

**No se preocupe si algunas clases le reportan errores (como marcas rojas) al importar el ejercicio por primera vez. Estos desaparecerán cuando termine correctamente los cambios requeridos en el mundo.**

4. Lea el enunciado del problema disponible en:

**n4\_parqueEmpresarial/docs/specs/Descripcion.pdf.**

5. Lea el documento de requerimientos funcionales disponible en:

**n4\_parqueEmpresarial/docs/specs/RequerimientosFuncionales.pdf.**

Identifique los nuevos requerimientos funcionales del ejercicio.

6. Explore el modelo del mundo diseñado para este ejercicio. Este modelo se encuentra en:

**n4\_parqueEmpresarial/docs/specs/ModeloConceptual.png.**

Identifique las clases, relaciones entre clases, atributos y métodos.

7. Revise el proyecto Java desde eclipse. Identifique los elementos del mundo (clases, atributos y métodos).

Se recomienda que consulte los siguientes videos:

- <https://youtu.be/KmhDGxPIGwI>. Este video le explicará cómo acceder al explorador de Eclipse.
- <https://youtu.be/tYxAI0F6s-c>. Este video le explicará cómo listar y acceder directamente a los métodos y atributos de una clase en forma rápida y eficiente.
- <https://youtu.be/Lnu9udEXTew>. Este video le explicará cómo dirigirse a la implementación (declaración) de una clase, método y atributo (objeto). Este conocimiento le será de gran utilidad para agilizar el desarrollo de su ejercicio y de parciales prácticos.

8. Desde Eclipse revise la documentación de las clases del mundo. Esto le permitirá entender para qué sirve cada método y cada uno de los atributos. El siguiente video explica cómo leer la documentación de un método:

<http://youtu.be/J0aQN39k7MA>

## Desarrollo

El desarrollo de este ejercicio se debe realizar a partir del ejercicio entregado en nivel 3. El esqueleto entregado solo tiene las clases nuevas del mundo. Para poder realizar el desarrollo,

debe copiar todas las clases del mundo del ejercicio de nivel 3. En el siguiente video se muestra cómo hacerlo:

<https://www.youtube.com/watch?v=CvaS-nRYLks>

## Parte 1: Clase Empleado

Completar los TODOs que comienzan por: **//TODO Parte1 PuntoY**

## Parte 2: Clase Empresa

En el archivo Empresa.txt encontrará los contratos y firmas de algunos de los métodos que debe completar o reemplazar en la clase Empresa. Recuerde reemplazar los contratos y las firmas de sus métodos con los proporcionados en el archivo.

1. Copie la clase Empresa desarrollada en el nivel 3 y péguela en la carpeta del mundo.
2. Cree todos los contratos faltantes de la clase y modifique los que considere necesario.
3. Elimine el atributo *numeroEmpleados* y declare y documente el nuevo atributo *empleados* como se muestra en el modelo conceptual.
4. Modifique la implementación del método constructor para que cumpla con el nuevo contrato dado en el archivo Empresa.txt.
5. Complete el método *darEmpleados* según la documentación dada en el archivo Empresa.txt.
6. Modifique el método *darNumeroEmpleados* para que se adapte a la nueva estructura de la clase Empresa, pero mantenga las condiciones de su contrato.
7. Complete el método *buscarEmpleado* según la documentación dada en el archivo Empresa.txt.
8. Declare y complete el método *agregarEmpleado* según la documentación dada en el archivo Empresa.txt.
9. Cree el contrato del método *eliminarEmpleado* que se encuentra en el archivo Empresa.txt.

## Parte 3: Clase Oficina



En el archivo Oficina.txt encontrará los contratos y firmas de algunos de los métodos que debe completar o reemplazar en la clase Oficina. Recuerde reemplazar los contratos y las firmas de sus métodos con los proporcionados en el archivo.

1. Copie la clase Oficina desarrollada en el nivel 3 y péguela en la carpeta del mundo.
2. Declare y documente las nuevas constantes de la clase, como se muestra en el modelo conceptual.
3. Modifique el método *ocuparOficina* para que cumpla con la documentación que se encuentra en el archivo Oficina.txt.
4. Modifique el método *desocuparOficina* para que cumpla con la documentación que se encuentra en el archivo Oficina.txt.
5. Copie y pegue en la clase Oficina el método *trabajaEnEmpresa* que se encuentra en el archivo Oficina.txt.
6. Declare e implemente el método *agregarEmpleado* para que cumpla con la documentación que se encuentra en el archivo Oficina.txt.
7. Copie y pegue en la clase Oficina el método *eliminarEmpleado* que se encuentra en el archivo Oficina.txt.

## Parte 4: Clase Piso

En el archivo Piso.txt encontrará los contratos y firmas de algunos de los métodos que debe completar de la clase Piso. Recuerde reemplazar los contratos y las firmas de sus métodos con los proporcionados en el archivo.

1. Copie la clase Piso desarrollada en el nivel 3 y péguela en la carpeta del mundo.
2. Modifique el método *ocuparOficina* para que cumpla con la documentación que se encuentra en el archivo Piso.txt.
3. Modifique el método *desocuparOficina* para que cumpla con la documentación que se encuentra en el archivo Piso.txt.
4. Copie y pegue en la clase Piso el método *buscarOficinaEmpleado* que se encuentra en el archivo Piso.txt.

5. Copiar y pegar el método *buscarOficinaEmpresa* que se encuentra en el archivo *Piso.txt*.
6. Crear el contrato del método *buscarOficinaEmpresa*.

## Parte 5: Clase Edificio

En el archivo *Edificio.txt* encontrará los contratos y firmas de algunos de los métodos que debe completar de la clase *Piso*. Recuerde reemplazar los contratos y las firmas de sus métodos con los proporcionados en el archivo.

1. Copie la clase *Edificio* desarrollada en el nivel 3 y péguela en la carpeta del mundo.
2. Copie y pegue en la clase *Edificio* el método *buscarOficinaEmpresa* que se encuentra en el archivo *Edificio.txt*.
3. Cree el contrato del método *buscarOficinaEmpresa*.
4. Modifique la firma y la implementación del método *ocuparOficinaEnPiso* para que cumpla con la documentación que se encuentra en el archivo *Edificio.txt*.
5. Modifique la firma y la implementación del método *desocuparOficina* para que cumpla con la documentación que se encuentra en el archivo *Edificio.txt*.
6. Copie y pegue en la clase *Edificio* el contrato y la firma del método *agregarEmpleado* que se encuentra en el archivo *Edificio.txt*.
7. Complete el método *agregarEmpleado* según la documentación dada.
8. Declare y complete el método *eliminarEmpleado* para que cumpla con la documentación que se encuentra en el archivo *Edificio.txt*.
9. Copie y pegue en la clase *Edificio* el método *registrarIngreso* que se encuentra en el archivo *Edificio.txt*.
10. Cree el contrato del método *registrarIngreso*.
11. Copie y pegue en la clase *Edificio* el método *registrarSalida* que se encuentra en el archivo *Edificio.txt*.

12. Cree el contrato del método *registrarSalida*

## Parte 6: Generación documentación

Genere la documentación de las clases, como se muestra en el siguiente video:

- Windows: [https://www.youtube.com/watch?v=vnF\\_NZdR1t4](https://www.youtube.com/watch?v=vnF_NZdR1t4)
- Mac: <https://www.youtube.com/watch?v=4ixjDpnEqQc>

## Validación

En esta sección se explica cómo validar el desarrollo del ejercicio. Para comprobar el completo y correcto funcionamiento de su ejercicio usted debe:

1. Ejecutar el programa e interactuar con todas las opciones disponibles en la interfaz. Los resultados obtenidos deben ser iguales a aquellos mostrados en el video demo.
2. Ejecutar las pruebas automáticas disponibles en el ejercicio. En el siguiente video <https://www.youtube.com/watch?v=rVd4AD8XMJk> se explica cómo efectuar esas pruebas. Alternativamente (por fuera de Eclipse), ejecute los programas: clean, cleanTest, build y luego buildTest que se encuentran en n4\_parqueEmpresarial/bin/win/ (para Windows) o en n4\_parqueEmpresarial /bin/mac/ (para MAC) y enseguida ejecute el programa runTest, haciendo doble clic en estos archivos. Estas pruebas deben presentar resultados en verde (0 errores y 0 fallas) cuando el ejercicio ha sido completado correctamente.
3. Tenga en cuenta que un error en un método diferente al método que está siendo probado puede afectar los resultados de la prueba.
4. Tenga en cuenta que las pruebas unitarias no son exhaustivas, y por ende es importante hacer las pruebas de interacción.

## Entrega

1. Indente el código fuente de todas las clases del mundo. En el siguiente enlace: <https://www.youtube.com/watch?v=cwQ9QiauaSc> encuentra un video que explica cómo indentar el código fuente de su ejercicio.
2. Construya el archivo entregable con su ejercicio desarrollado y validado completamente. En el siguiente video: <https://www.youtube.com/watch?v=xuSDFfEZW78> se explica detalladamente el proceso para producir el archivo comprimido del ejercicio y enviarlo a

SicuaPlus. No olvide revisar que su entrega cumple con lo especificado en las normas del curso (presentes en el siguiente enlace:

<http://cupi2.uniandes.edu.co/sitio/index.php/cursos/apo1/normas-administrativas>).

Esto incluye renombrar el archivo a entregar con su login de la siguiente forma:

**n<nivel del ejercicio>\_<login estudiante>.zip**

**Por ejemplo: n4\_tsuarez.zip**

**La no indentación del código fuente o el nombramiento incorrecto del ejercicio en su entrega es una acción penalizada en la plantilla de calificación del mismo.**

3. Entregue el archivo del ejercicio vía SicuaPlus, de acuerdo con las normas, fecha y hora de entrega