

Práctica: aplicación para gestionar cultivos de bacterias

La práctica evaluará sobre 10 puntos sin la implementación de swing, aunque se valorará de manera positiva la entrega con swing (+2 puntos sobre 10).

Para realizar un conjunto de experimentos en un laboratorio de biólogos necesitan una aplicación de escritorio que les ayude a recoger la información de los experimentos. Los experimentos consisten en realizar cultivos de bacterias. En cada experimento se parte de una población de bacterias inicial. En el medio donde viven las bacterias se añade una determinada cantidad de comida. Esta cantidad de comida inicial irá incrementando linealmente durante varios días, y a continuación disminuye linealmente hasta un valor final. Todos los experimentos duran 30 días.

La aplicación deberá permitir gestionar la información relativa a las poblaciones de bacterias asociadas con un experimento. Cada experimento se guardará en un archivo diferente, y cada experimento puede estar compuesto por múltiples poblaciones de bacterias. Para cada población de bacterias, inicialmente el científico indicará: un nombre asociado a esa población, la fecha de comienzo de la población, la fecha de fin de la población, el número de bacterias iniciales, la temperatura a la cual se van a someter las bacterias, las condiciones de luminosidad ("Alta", "Media" y "Baja") y la dosis de comida. Para indicar la dosis de comida, los biólogos deben introducir la cantidad inicial de comida que se le dará el primer día, el día (de entre los 30 días que duran las poblaciones) hasta el cual se debe incrementar la cantidad de comida y la comida de este día, y la cantidad final de comida en el día 30. A partir de estos datos, la aplicación irá calculando la cantidad de comida que hay que proporcionar cada día a las bacterias. Las cantidades de comida son valores enteros menores que 300.

La aplicación deberá permitir realizar las siguientes acciones:

1. Abrir un archivo que contenga un experimento
2. Crear un nuevo experimento
3. Crear una población de bacterias y añadirla al experimento actual
4. Visualizar los nombres de todas las poblaciones de bacterias del experimento actual
5. Borrar una población de bacterias del experimento actual
6. Ver información detallada de una población de bacterias del experimento actual
7. Guardar (se supone que para usar esta opción previamente hemos abierto un archivo)
8. Guardar como

Cuando el usuario seleccione la opción 6, la aplicación deberá solicitar el nombre de la población de la cual quiere ver los detalles, y a continuación deberá mostrarse toda la información detallada del experimento, inclusive la dosis de alimento que se debe dar a las bacterias cada uno de los 30 días.

Necesariamente habrá que guardar en modo texto toda la información correspondiente con cada experimento.

Consejos

Habla con el profesor para determinar cuál es la mejor forma para representar y almacenar los datos, y de estructurar la práctica. Organiza la práctica en al menos tres paquetes. Uno se encargará de la interfaz de usuario, otro se encargará de cargar y guardar datos en el disco duro, y el tercero se encargará de la lógica de negocio: añadir ratones, eliminar ratones, etc.... Utiliza funciones y métodos aislados, atómicos y con una responsabilidad completa. Si tienes dudas sobre la utilidad de una función o método, consulta con el profesor. Visualice especialmente las clases donde se realiza el UML y la separación de paquetes.

Sobre la entrega y evaluación de la práctica

La práctica debe ser entregada en formato electrónico antes del 26 de marzo (inclusive). Para considerar que la práctica está entregada, el alumno deberá haber completado la entrega en formato electrónico (a través del campus virtual). **La práctica deberá contener así mismo un fichero ejecutable .jar que permita ejecutar el código fuente sin necesidad de compilar el código. Si no se encuentra el fichero .jar la práctica se calificará con un 0.**

Por cada semana que se retrase la entrega de la práctica a partir de esta fecha, la nota final de la práctica se le descontará 1 punto (0.15 puntos por día aproximadamente). La práctica puede realizarse a través de una consola o a través de una aplicación gráfica de escritorio empleando swing. El alumno tendrá que realizar una defensa de la práctica; si el alumno es incapaz de defender la práctica y explicar su funcionamiento, su nota en la práctica será un 0.

Ítems a entregar: Implementación íntegra de los requisitos establecidos en el enunciado de la práctica junto a la memoria descriptiva del trabajo realizado. Se evaluará de forma más positiva la descripción de requisitos no implementados y conocidos descritos en la memoria que la falta de funcionalidad desconocida y no descrita.

Todas las clases y sus clases y funciones contenidas deben tener las correspondientes pruebas unitarias.

Como resultado de esta entrega final se deberá enviar un fichero **zip** con la carpeta completa que contenga el proyecto software **IntelliJ o NetBeans** correspondiente a la práctica (asegúrese de que incluye todo el código fuente, es la carpeta anterior al src, test, etc..) y además un fichero ejecutable .jar que permita ejecutar la práctica sin tener que compilar el código fuente.

La entrega de la práctica forma parte de la evaluación de la práctica.

La memoria incluirá una portada con el nombre del alumno y se ajustará a la siguiente estructura:

- Análisis y descripción de la aplicación. Este análisis y descripción dará respuesta a las siguientes preguntas:
 - o Cómo se han organizado y estructurado las clases y cuál es la responsabilidad de cada una.
 - o Qué decisiones de diseño se han tomado.
 - o Qué comprobaciones de integridad (y excepciones) se han implementado.
 - o Qué técnicas de ordenación y búsqueda ha utilizado y por qué lo ha hecho.
 - o Diagramas de clases UML.
- **Listado de fallos conocidos y funcionalidades definidas en el enunciado que no se han implementado** en el código entregado.
- Listado de todo el código fuente de la aplicación organizado por paquetes (si aplica) y clases.
- Listado de pruebas unitarias realizadas.
- Conclusiones (que incluirán, obligatoriamente, una valoración del tiempo dedicado a la práctica).

Además de la defensa, la práctica se evaluará en relación a:

- **Organización y estructura del código** (utilización de conceptos y patrones de programación, orientada a objetos: herencia, polimorfismo, encapsulación, reutilización, utilización correcta de estructuras de control, etc.).
- **Uso de Javadoc** para documentar las clases y métodos.
- Funcionamiento ajustado a los requisitos establecidos (incluyendo, además de chequeos de datos, integridad de la información, gestión de excepciones...).
- **Claridad del código y adecuación a las normas de estilo** (correcto nombrado de clases, métodos y variables, comentarios internos, indentación del código...).

Se advierte al alumno de que, como parte de la evaluación, se utilizará una herramienta antiplagio de código. En caso de copia total o parcial del código de la práctica, se aplicarán las medidas correspondientes en el código de conducta de la universidad CEU-San Pablo.