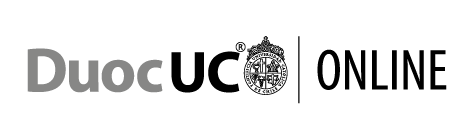
Rectángulo: esquinas redondeadas 1, Forma

Semana 6

Desarrollo Orientado a Objetos I (PRY2202)

Pauta de evaluación sumativa

# Aplicando Entradas y salidas (I/O) y Manejo de Archivos

A continuación, se presentan los descriptores de los niveles de logro con los que serás retroalimentado/a y/o evaluado/a:

| **CL** | Completamente Logrado (100%) | Logro **completo** del manejo del indicador, se considera como el punto **óptimo**, de acuerdo al área evaluada, dentro del rango de competente |
| --- | --- | --- |
| **L** | Logrado (80%) | Logro del indicador, se observan **pequeñas dificultades o errores** para alcanzar el logro total y sobresaliendo de lo mínimo aceptable |
| **ML** | Medianamente Logrado (60%) | Logro **mínimo aceptable** en el manejo del indicador que permite ser considerado competente |
| **LI** | Logro Insuficiente (30%) | **Logro insuficiente** en el manejo del indicador, se observan **varias dificultades o errores** que no le permiten alcanzar el logro mínimo para ser considerado competente |
| **NL** | No Logrado (0%) | Se observan un **escaso, nulo o incorrecto** logro en el manejo del indicador |

| **N°** | **Criterios** | **Completamente Logrado (100%)** | **Logrado (80%)** | **Medianamente Logrado (60%)** | **Logro Insuficiente** **(30%)** | **No Logrado** **(0%)** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | Implementa colecciones ‘ArrayList y ‘HashMap’ para almacenar y gestionar objetos.  **(15 puntos)** | Implementa eficazmente ‘ArrayList’ con operaciones avanzadas y ‘HashMap’ con gestión óptima de claves-valor. | Utiliza correctamente ArrayList y HashMap para almacenar y gestionar objetos, aplicando métodos adecuados para operaciones comunes. | Implementa ‘ArrayList’ correctamente, pero ‘HashMap’ tiene errores. | Implementa ‘ArrayList’ y ‘HashMap’ con errores menores y funcionalidad básica. | Implementa ‘ArrayList’ y ‘HashMap’ incorrectamente. |
| 2 | Integra clases ‘HashSet’ y ‘TreeSet’ para manejar colecciones de datos.  **(15 puntos)** | Integra ‘HashSet’ y ‘TreeSet’ para necesidades específicas optimizando unicidad y ordenación. | Implementa bien ‘HashSet’ y ‘TreeSet’ con algunas optimizaciones. | Implementa bie ‘HashSet’ y ‘TreeSet’ pero sin optimizaciones. | Implementa ‘HashSet’ y ‘TreeSet’ de forma básica. | No utiliza ‘HashSet’ y ‘TreeSet’ adecuadamente. |
| 3 | Implementa bloques ‘try/catch’ para el manejo de excepciones.  **(15 puntos)** | Implementa bloques try/catch de forma exhaustiva, manejando todas las excepciones anticipadas con acciones de recuperación específicas y registro detallado de errores. | Implementa bloques try/catch correctamente para la mayoría de las operaciones críticas, con acciones de recuperación adecuadas y algún registro de errores. | Implementa bloques try/catch para las operaciones esenciales, identificando y manejando las excepciones comunes sin detalles de recuperación específicos. | Implementa bloques try/catch básicos en puntos críticos del programa, pero con manejo de excepciones genérico y limitada recuperación de errores. | Implementa pocos o nulos bloques try/catch, dejando varias operaciones críticas sin manejo de excepciones o con manejo inadecuado. |
| 4 | Integra archivos externos de CSV.  **(15 puntos)** | Integra archivos externos de CSV excepcionalmente, destacando por su eficiencia y manejo óptimo de los datos. | Integra archivos CSV de manera competente y con mínimos errores apreciables. | Integra archivos CSV de manera funcional, pero podría mejorarse. | Implementa la integración de archivos CSV, pero con errores o limitaciones significativas. | No implementa la integración de archivos externos de CSV. |
| 5 | Lee datos externos con ‘BuffererdReader’ para la creación de objetos  **(20 puntos)** | Lee datos de archivos CSV sin errores, creando objetos ‘Libro eficientemente. | Lee datos de archivos CSV eficazmente con mínimos errores y creando objetos ‘Libro’ competentes y eficientes, manejando adecuadamente los datos leídos. | Lee datos de archivos CSV con eficacia, con algunos errores menores, y creando objetos ‘Libro’ funcionales que podrían mejorarse en eficiencia. | Lee datos de archivos CSV con errores significativos y en la creación de objetos ‘Libro’, con implementaciones incompletas o incorrectas. | Falla en leer datos de archivos CSV adecuadamente y no logra crear objetos ‘Libro’. |
| 6 | Escribe datos externos con ‘FileWriter’ para guardar la información de los usuarios en archivos de texto.  **(20 puntos)** | Escribe datos de usuarios en archivos de texto sin errores, con un formato claro y estructurado, manejando excepciones de manera eficaz. | Escribe datos de usuarios eficientemente, con mínimos errores de formato o manejo de excepciones. | Escribe datos de usuarios en archivos de texto con eficacia, pero con margen de mejora en formato o manejo de excepciones. | Escribe datos de usuarios con errores significativos en formato o manejo inadecuado de excepciones. | Intenta escribir datos de usuarios en archivos de texto, pero falla debido a errores graves o manejo ineficaz de excepciones. |



Reservados todos los derechos Fundación Instituto Profesional Duoc UC. No se permite copiar, reproducir, reeditar, descargar, publicar, emitir, difundir, de forma total o parcial la presente obra, ni su incorporación a un sistema informático, ni su transmisión en cualquier forma o por cualquier medio (electrónico, mecánico, fotocopia, grabación u otros) sin autorización previa y por escrito de Fundación Instituto Profesional Duoc UC La infracción de dichos derechos puede constituir un delito contra la propiedad intelectual.