**Carpeta: PYxVanessa**

**TAREAS REALES POR APLICAR EN VANESSA**

* **Automatización de copias de seguridad (.BAK)**

Tu siguiente reto como Inge. será desarrollar un script en Python que **automatice el proceso de respaldo y verificación de archivos de copia de seguridad (.BAK)** utilizados por el sistema.

Este desarrollo será parte de las tareas reales de mantenimiento del servidor de la empresa y deberá ser robusto, claro y seguro.

**Objetivo:**

El sistema deberá verificar si existen respaldos .BAK en una carpeta origen, validar que el destino tenga suficiente espacio disponible en disco para recibirlos, mover los archivos de forma segura y registrar todo lo ocurrido en un archivo de log.

Además, en caso de espacio insuficiente, deberá intentar liberar espacio eliminando los archivos más antiguos del destino. Si no es posible liberar espacio, deberá registrar claramente la causa y evitar pérdidas de información.

**Nota:** Antes de programar nada, escribe en una hoja una **propuesta clara y ordenada** sobre como resolverías el problema de automatizar las copias de seguridad.

**📥 Entrada esperada:**

El script deberá trabajar con las siguientes rutas configurables:

* Carpeta **origen** de los archivos .BAK
* Carpeta **destino** de almacenamiento
* Archivo **log** donde se documentará el proceso cada día

**No olvides** que tu valor como desarrollador está en diseñar soluciones útiles y seguras, no sólo en escribir código.

**🔎 Antes de programar: tu reto**

Antes de escribir una sola línea de código, debes **analizar y diseñar** cómo resolverías este problema de forma lógica, ordenada y profesional.

Responde cada punto a continuación con claridad:

* 1️⃣ Comprensión del problema
  + ¿Cuál es el propósito real de este script en la empresa?
  + ¿Qué riesgos podrían existir si se hace mal este proceso?
* 2️⃣ Identificación de condiciones clave
  + ¿Qué situaciones especiales debes tener en cuenta para evitar errores graves? Ejemplos:
    - ¿Qué pasa si no hay archivos?
    - ¿Qué pasa si no hay espacio en disco?
    - ¿Qué pasa si un archivo está en uso?
* 3️⃣ Diseño del flujo de proceso
  + Redacta paso a paso como debería comportarse tu script. No uses código**,** solo **descripción lógica.**
    - Por ejemplo: Verificar si hay archivos en el origen, si no hay, registrar el mensaje en el log y finalizar
* 4️⃣ Consideraciones técnicas
  + ¿Cómo validarás el espacio disponible en disco?
  + ¿Qué criterio usarás para eliminar archivos antiguos si no hay espacio?
  + ¿Cómo registrarás el resultado de cada acción (éxito o fallo)?
* 5️⃣ Diseño del archivo de log
  + ¿Qué formato tendrá el log? ¿Qué datos incluirá por cada evento?
  + ¿Cómo identificarás fácilmente si el proceso tuvo errores?
* 6️⃣ Escenarios especiales

Propón al menos **3 situaciones problemáticas** que tu script debe saber manejar.  
Ejemplo:

* + Hay espacio, pero el archivo está siendo usado por otra aplicación.
  + Se copia un archivo, pero no se verifica que se copió bien.
  + No se puede eliminar un archivo antiguo por permisos.
* 💡 Pistas
  + Para ayudarte en tu diseño lógico, ten en cuenta que podrías usar herramientas como:
    - **os** y **shutil** para mover archivos.
    - **psutil** para medir espacio en disco.
    - **logging** o escritura en .txt para los logs.
  + Pero **no te concentres en eso aún**. Solo piensa en cómo debería funcionar **idealmente** el sistema.

RESPUESTA

* 1️⃣Automatización en el proceso de respaldo y verificación de archivos de las copias de seguridad .BAK garantizando la disponibilidad para la recuperación de archivos, así como un uso eficiente del espacio en disco eliminando backups antiguos y que registre cada específicamente cada proceso en un log.
* El propósito de este script es automatizar el proceso de respaldo y verificación de archivos .BAK asegurando que las copias de seguridad sean válidas, accesibles y recientes, mientras se gestiona el espacio en disco.
* Desaparición del archivo
* Errores de restauración
* corrupción del archivo
* Archivos desactualizados
* Eliminación de archivos validos
* Bloqueo de archivos
* 2️⃣
* Si el espacio del disco está lleno entonces el sistema lo detecta y busca el/los Backup más antiguo/s y los elimina para priorizar la nueva copia, la antigüedad se valida por la fecha de modificación.
* Si el archivo se encuentra en uso en el momento que se vaya a generar el Backup el sistema omite el archivo y continua el proceso con los demás archivos.
* Ahora si en el momento del Backup no se encuentra ningún archivo entonces el proceso se finaliza.
* Nunca borrar archivos del mismo día
* conservar al menos los últimos 30 archivos

NOTA: Cada uno de estos eventos/procesos debe tener su registro

* 3️⃣

1. Entrar y validar que la carpeta Origen exista y nos permita usarla
2. Buscar archivos .BAK y validar que si existan y devolver una lista de los archivos
3. Registrar cada evento según su acción.
4. Validar que la carpeta Destino cuente con espacio suficiente para los backups
5. Liberar espacio eliminando los backups más antiguos hasta contar con el espacio necesario para los nuevos backups
6. Procesar cada archivo:  
   - Validar que los archivos no estén bloqueados o en uso  
   - Validar el tamaño del archivo, que pese más de 0 MB

- Chequear que el archivo no haya sido modificado recientemente

- Si el archivo es válido, moverlo a la carpeta de Destino

- Que finalice la ejecución

* 4️⃣

1. El sistema consultará el espacio libre en la unidad de almacenamiento de Destino. Función clave: psutil.disk\_usage(path) devuelve el uso del disco en el directorio especificado (path), incluyendo el espacio total, usado y libre.
2. Tener en cuenta las fechas de modificación del/los archivos y eliminar los archivos mas antiguos hasta liberar por completo el espacio necesario.
3. Cada evento del proceso se documentará en un archivo de log estructurado.

* 5️⃣

Formato de Log

1. Fecha y hora exactas
2. Tipo: Info, Error, Success, Warning. Etc
3. Evento
4. Nombre del archivo
5. Ruta del archivo
6. Tamaño del archivo

Texto

El contenido generado por IA puede ser incorrecto.

Revisión por tipo de evento en el log

Se buscarán entradas con el tipo ERROR o WARNING.