**Parte 1:**

Caso 1

* Full: Dom 9hs Semanal
* Diferencial: Diario 22hs (puede ser en dos momentos del día)
* Log: Cada 30 min

Caso 2

* Full: dom 1hs cada 2 semanas
* Diferencial: Diario 1hs cada 2 días
* Log: Cada 10 min.

Ejemplo:

Muere martes 9:05hs

1. Full Dom
2. Diferencial Lunes
3. Todos los log hasta el martes a las 9. Se pierden 5 minutos de trabajo.

-- Poner ejemplo en el parcial y aclarar lo más que se pueda. Suma puntos.

**Parte 2: Integridad**

Caso 3

1-a) Poner el campo código de producto como clave primaria.

1-b) El detalle debe estar relacionado con la tabla cabecera. Clave extranjera.

1-c) Control de campo no nulo al ingresar el dato.

1-d) Restricción de campo para sólo ingresar números.

1-e) Primero eliminar los tickets asociados al cliente y luego el cliente. Clave foránea.

1-f) Si la cantidad de productos asociados a un proveedor es mayor a 10, disparar un evento para que los datos del proveedor sean de sólo lectura. Trigger porque se ejecuta automáticamente cuando se ejecuta un evento.

1-g) Al ingresar la fecha de emisión, hacer una validación de campo. Si es menor a 2010, no ingresarlo. Check.

**Parte 3: Concurrencia**

1. Ambos clientes leen datos, por lo que no surge ningún error de concurrencia, pero sí es concurrencia.
2. Como la transacción que hizo el primer cliente no terminó, el segundo cliente no puede consultar el precio unitario de ese mismo producto hasta que no se ejecute el COMMIT. Esta consulta queda pendiente.

Problema de bloqueo: El bloqueo de escritura bloquea al de lectura. Los de aproximación no se bloquean mutuamente.

1. Registro: 1:89:3
2. El nivel de aislamiento está colocado en READ UNCOMMITED (en SQL se baja 1 nivel con NOLOCK), es decir, que se ven los cambios que todavía no se confirmaron (problema de lectura sucia). Es recomendable utilizar un nivel mayor. SQL usa por defecto READ COMMITED.

No dejar transacciones abiertas. Usar transacciones cortas (BEGIN y COMMIT en botón aceptar) complementado con una técnica optimista.