Casos de Prueba - Tarea Integradora 1

## Funcionalidad: RF1 - Registrar Ruta

Escenario: Ingresar ruta con datos válidos

Resultado esperado: Ruta creada exitosamente.

## Funcionalidad: RF1 - Registrar Ruta

Escenario: Ingresar ruta con distancia negativa

Resultado esperado: Debe lanzar IllegalArgumentException.

## Funcionalidad: RF2 - Registrar Incidente

Escenario: Ingresar incidente con todos los campos válidos

Resultado esperado: Incidente registrado correctamente.

## Funcionalidad: RF2 - Registrar Incidente

Escenario: Ingresar incidente con descripción vacía

Resultado esperado: Debe lanzar IllegalArgumentException.

## Funcionalidad: RF3 - Registrar Conductor

Escenario: Ingresar conductor con datos válidos

Resultado esperado: Conductor registrado correctamente.

## Funcionalidad: RF3 - Registrar Conductor

Escenario: Ingresar conductor con nombre vacío

Resultado esperado: Debe lanzar IllegalArgumentException.

## Funcionalidad: RF4 - Buscar Conductor

Escenario: Buscar conductor existente por nombre

Resultado esperado: Debe retornar el objeto conductor.

## Funcionalidad: RF4 - Buscar Conductor

Escenario: Buscar conductor no existente

Resultado esperado: Debe lanzar ConductorNoEncontradoException.

## Funcionalidad: RF5 - Buscar Incidente

Escenario: Buscar incidente existente por ID

Resultado esperado: Debe retornar el objeto incidente.

## Funcionalidad: RF5 - Buscar Incidente

Escenario: Buscar incidente inexistente por ID

Resultado esperado: Debe lanzar IncidenteNoEncontradoException.

## Funcionalidad: RF6 - Ordenar Rutas

Escenario: Ordenar rutas por distancia

Resultado esperado: Lista ordenada correctamente.

## Funcionalidad: RF6 - Ordenar Rutas

Escenario: Ordenar rutas por tiempo estimado

Resultado esperado: Lista ordenada correctamente.

## Funcionalidad: RF7 - Ordenar Incidentes

Escenario: Ordenar incidentes por fecha

Resultado esperado: Incidentes ordenados del más reciente al más antiguo.

## Funcionalidad: RF7 - Ordenar Incidentes

Escenario: Ordenar incidentes con fechas iguales

Resultado esperado: Mantiene el orden de entrada o por ID.

## Funcionalidad: RF9 - Guardar/Cargar JSON

Escenario: Guardar y cargar datos de conductores

Resultado esperado: Datos deben persistir correctamente.

## Funcionalidad: RF9 - Guardar/Cargar JSON

Escenario: Cargar cuando no hay archivos

Resultado esperado: Debe retornar listas vacías sin error.