



CLASE PACIENTE: PRUEBAS UNITARIAS

Realizar las pruebas unitarias para la clase Paciente para asegurar su correcto funcionamiento y verificar que todas las funcionalidades proporcionadas se ejecutan como se espera.

Al menos estas:

- Realiza una prueba para verificar que la función saludar() devuelve el saludo correcto.
- Prueba las funciones de obtener y modificar para nombre, apellidos, y fecha de nacimiento. Asegúrate de que las modificaciones se reflejen correctamente.
- Verifica con una prueba que la función obtenerEdad() devuelve la edad correcta basada en la fecha de nacimiento (para calcular la edad usar sólo el año).
- Realiza una prueba para asegurarte de que la función calcularIMC() devuelve un número.
- Puedes realizar pruebas adicionales si consideras necesario, por ejemplo, para las funciones modificarBascula() y obtenerBascula().

Criterios de éxito:

- Todas las pruebas deben pasar sin errores.
- Las pruebas deben cubrir todos los métodos de la clase Paciente.
- Las pruebas deben ser claras y descriptivas en cuanto a lo que están verificando.

INSTRUCCIONES DE AYUDA:

Para comparar una fecha (objeto tipo Date), con una fecha tipo "texto" podemos usar esta instrucción de JavaScript:

- `[objeto DATE].toISOString().split('T')[0].to.equal('1990-11-02');`
Al convertir una fecha en cadena obtenemos algo así: 1980-11-02T00:00:00.000Z
"partimos (split)" y nos quedamos con el primer item

Dentro de la clase "Paciente" tenéis más "usos" para fechas.



CLASE ANALIZADORDESANGRE: TDD

Desarrollar la clase `AnalizadorDeSangre` utilizando un enfoque TDD basado en las pruebas proporcionadas.

Preparación: Familiarízate con las pruebas dadas para la clase `AnalizadorDeSangre`. Estas pruebas serán tu guía para desarrollar la funcionalidad requerida.

Iteración TDD:

- Comienza por ejecutar las pruebas. Dado que aún no has escrito la clase `AnalizadorDeSangre`, las pruebas fallarán.
- Desarrolla la mínima cantidad de código necesario para hacer pasar la primera prueba.
- Ejecuta las pruebas nuevamente. Si alguna prueba no pasa, corrige tu código y repite el proceso.
- Continúa con la siguiente prueba y repite el proceso hasta que todas las pruebas pasen.

Requisitos de la clase `AnalizadorDeSangre`:

- La clase debe tener un atributo para almacenar los últimos resultados del análisis de sangre (se almacenarán en un objeto, `{}`, en el que posteriormente añadiremos "colesterol ,hierro,...")
- Debe tener un método que genera valores aleatorios entre un rango dado (min y max). Para generar valores aleatorios: `Math.random() * (max - min) + min;`
- Debe tener un método para analizar una muestra de sangre, el cual generará valores aleatorios para parámetros como el colesterol y el hierro.
- Debe tener un método que devuelva los últimos resultados del análisis de sangre.
- Debe tener un método que determine si el nivel de colesterol es alto, considerando un colesterol alto como un valor superior a 240.