

Notas profesor

Caso 1

Imagínate que eres el profesor de una asignatura y has puesto a lo largo del trimestre tres exámenes, y quieres calcular la nota media de tus alumnos.

Tenemos los siguiente datos

Alumno	Nota 1	Nota 2	Nota 3
Juan Pérez	7	8	9
María López	6	7	8
Andrés Ruiz	9	9	10
Laura Gómez	8	7	6
Carlos Sánchez	3	2	5
Ana Martínez	10	9	10
Sergio Torres	7	8	6
Isabel Rodríguez	9	10	9

Vamos a empezar definiendo la firma del método para calcular la nota media de los alumnos:

```
const calcularNotaMediaAlumnos = (fichaAlumno: FichaAlumno[]): Notas[] => {  
  // TODO  
};
```

Parámetros de entrada

El parámetro de entrada es un array de objetos de tipo `FichaAlumno` que tiene dos propiedades:

- `alumno`: que es un string con el nombre del alumno
- `notas`: que es un array de números con las notas de los exámenes

Creamos su interfaz:

```
interface FichaAlumno {  
  alumno: string;  
  notas: number[];  
}
```

Parámetros de salida

Como salida vamos a devolver un array de objetos de tipo `Notas` que tiene dos propiedades:

- `alumno`: que es un string con el nombre del alumno.
- `notaMedia`: que es un número con la nota media.

Creamos su interfaz:

```
interface Notas {  
  alumno: string;  
  notaMedia: number;  
}
```

Lo primero que vamos a hacer es crearnos un método para calcular la nota media de un alumno:

```
const calcularNotaMedia = (notas: number[]): number => {  
  // TODO  
};
```

Parámetros de entrada

El parámetro de entrada es un array de números con las notas de los exámenes.

Parámetros de salida

Como salida vamos a devolver un número con la nota media.

Vamos a implementar una batería de pruebas para el método `calcularNotaMedia`:

```
describe("calcularNotaMedia", () => {  
  it("debería devolver la nota media de un alumno", () => {  
    // Arrange  
    const notas: number[] = [7, 8, 9];  
  
    // Act  
    const result = calcularNotaMedia(notas);  
  
    // Assert  
    const notaMedia = 8;  
  
    expect(result).toEqual(notaMedia);  
  });  
  
  it("debería devolver la nota media de un alumno", () => {  
    // Arrange  
    const notas: number[] = [6, 7, 8];
```

```

    // Act
    const result = calcularNotaMedia(notas);

    // Assert
    const notaMedia = 7;

    expect(result).toEqual(notaMedia);
  });

  it("debería devolver la nota media", () => {
    // Arrange
    const notas: number[] = [3, 2, 6];

    // Act
    const result = calcularNotaMedia(notas);

    // Assert
    const notaMedia = 3.67;

    expect(result).toEqual(notaMedia);
  });
});

```

Te toca, intenta implementar el método `calcularNotaMedia`:

Pistas:

- Usar el método `reduce` para sumar todas las notas.
- Usar el método `length` para saber cuántas notas hay.
- Usar el método `toFixed` para redondear la nota media a dos decimales.

Sino has encontrado la solución, no desesperes, lo importante es haberlo intentado, aquí tienes la solución:

```

export const calcularNotaMedia = (notas: number[]): number => {
  const sumaNotas = notas.reduce((acc, nota) => acc + nota, 0);
  const notaMedia = sumaNotas / notas.length;
  const notaMediaRedondeada = Number(notaMedia.toFixed(2));

  return notaMediaRedondeada;
};

```

Ahora que ya podemos calcular la nota media de un alumno, vamos a añadirle al método `calcularNotaMediaAlumnos` unas pruebas para que nos sirvan de guía para implementar el método 😊:

```

describe("calcularNotaMediaAlumnos", () => {
  it("debería devolver la nota media de todos los alumnos", () => {
    // Arrange
    const fichaAlumnos: FichaAlumno[] = [
      {

```

```
    alumno: "Juan Pérez",
    notas: [7, 8, 9],
  },
  {
    alumno: "María López",
    notas: [6, 7, 8],
  },
];

// Act
const result = calcularNotaMediaAlumnos(fichaAlumnos);

// Assert
const notasMedias: Notas[] = [
  {
    alumno: "Juan Pérez",
    notaMedia: 8,
  },
  {
    alumno: "María López",
    notaMedia: 7,
  },
];

expect(result).toEqual(notasMedias);
});
});
```

Te toca, implementar el método `calcularNotaMediaAlumnos`.

Pistas:

- Usar el método `map` para recorrer el array de `fichaAlumnos`.
- Usar el método `calcularNotaMedia` para calcular la nota media de cada alumno.

Sino encuentras la solución, no desesperes, aquí la tienes:

```
export const calcularNotaMediaAlumnos = (
  fichaAlumnos: FichaAlumno[]
): Notas[] => {
  const notasMedias: Notas[] = fichaAlumnos.map((fichaAlumno) => {
    const notaMedia = calcularNotaMedia(fichaAlumno.notas);

    return {
      alumno: fichaAlumno.alumno,
      notaMedia,
    };
  });

  return notasMedias;
};
```

Caso 2

Este profesor además tiene una parte de ejercicios prácticos, la nota de los ejercicios prácticos pondera un 60% sobre la nota final, y los exámenes un 40%.

Calcula la nota final de los alumnos, si ha tenido los siguientes resultados:

Alumno	Practica 1	Practica 2	Nota 1	Nota 2	Nota 3
Juan Pérez	7	8	7	8	9
María López	3	0	6	7	8
Andrés Ruiz	8	9	9	9	10
Laura Gómez	10	10	8	7	6
Carlos Sánchez	5	7	6	5	7
Ana Martínez	6	8	10	9	10
Sergio Torres	6	7	7	8	6
Isabel Rodríguez	8	10	9	10	9

Vamos a empezar definiendo un método donde nos devuelva un objeto para cada alumno con su nombre y la nota final, esta sería la firma del método:

```
const calcularCalificacionAlumnos = (  
  fichaAlumno: FichaFinalAlumno[]  
) : CalificacionesAlumno[] => {  
  // TODO  
};
```

Parámetros de entrada

El parámetro de entrada es un array de objetos de tipo `FichaFinalAlumno` que tiene tres propiedades:

- `alumno`: que es un string con el nombre del alumno
- `notaPracticas`: que es un array de números con las notas de los ejercicios prácticos
- `notaExamen`: que es un array de números con las notas de los exámenes

Creemos su interfaz:

```
interface FichaFinalAlumno {  
  alumno: string;  
  notasPracticas: number[];  
  notasExamen: number[];  
}
```

Parámetros de salida

Como salida vamos a devolver un array de objetos de tipo `CalificacionesAlumno` que tiene dos propiedades:

- `alumno`: que es un string con el nombre del alumno.
- `notaFinal`: que es un número con la nota final.

Creamos su interfaz:

```
interface CalificacionesAlumno {  
  alumno: string;  
  notaFinal: number;  
}
```

En el caso anterior teníamos un método para calcular la nota media de los alumnos, `calcularNotaMedia`, pero este mismo método nos va a servir para calcular la nota media de los ejercicios prácticos, ya que la única diferencia es que en vez de pasarle las notas de los exámenes, le pasamos las notas de los ejercicios prácticos.

Lo siguiente que vamos es crear un método que nos calcule si la parte práctica de la asignatura tiene un peso en la nota final del 60% y los exámenes un 40%:

```
const calcularNotaFinal = (  
  notaMediaPracticas: number,  
  notaMediaExamenes: number  
): number => {  
  // TODO  
};
```

Parámetros de entrada

Los parámetros de entrada son dos números:

- `notaMediaPracticas`: que es un número con la nota media de los ejercicios prácticos.
- `notaMediaExamenes`: que es un número con la nota media de los exámenes.

Parámetros de salida

Como salida vamos a devolver un número con la nota final.

Vamos a implementar una batería de pruebas para el método `calcularNotaFinal`:

```
describe("calcularNotaFinal", () => {  
  it("debería devolver la nota final", () => {  
    // Arrange  
    const notaMediaPracticas = 8;
```

```
    const notaMediaExamenenes = 7;

    // Act
    const result = calcularNotaFinal(notaMediaPracticass, notaMediaExamenenes);

    // Assert
    const notaFinal = 7.6;

    expect(result).toEqual(notaFinal);
  });

  it("debería devolver la nota final", () => {
    // Arrange
    const notaMediaPracticass = 6;
    const notaMediaExamenenes = 7;

    // Act
    const result = calcularNotaFinal(notaMediaPracticass, notaMediaExamenenes);

    // Assert
    const notaFinal = 6.4;

    expect(result).toEqual(notaFinal);
  });

  it("debería devolver la nota final", () => {
    // Arrange
    const notaMediaPracticass = 3;
    const notaMediaExamenenes = 2;

    // Act
    const result = calcularNotaFinal(notaMediaPracticass, notaMediaExamenenes);

    // Assert
    const notaFinal = 2.6;

    expect(result).toEqual(notaFinal);
  });
});
```

Te toca, intenta implementar el método `calcularNotaFinal`:

Pistas:

- Para calcular la nota final, multiplicamos la nota media de los ejercicios prácticos por 0.6 y la nota media de los exámenes por 0.4.
- Usar el método `toFixed` para redondear la nota final a dos decimales.

Sino has encontrado la solución, no desesperes, aquí la tienes:

```
export const calcularNotaFinal = (
  notaMediaPracticass: number,
```

```

    notaMediaExamen: number
  ): number => {
    const notaFinal = notaMediaPracticas * 0.6 + notaMediaExamen * 0.4;
    const notaFinalRedondeada = Number(notaFinal.toFixed(2));

    return notaFinalRedondeada;
  };

```

Una vez que ya hemos creado el método para calcular la nota final, vamos a añadirle al método `calcularCalificacionAlumnos` unas pruebas para que nos sirvan de guía para implementar el método ☺:

```

describe("calcularAlumnosNotaFinal", () => {
  it("debería devolver la nota final de todos los alumnos", () => {
    // Arrange
    const fichaFinalAlumnos: FichaFinalAlumno[] = [
      {
        alumno: "Juan Pérez",
        notasPracticas: [7, 8],
        notasExamen: [7, 8, 9],
      },
      {
        alumno: "María López",
        notasPracticas: [3, 0],
        notasExamen: [6, 7, 8],
      },
    ];

    // Act
    const result = calcularCalificacionAlumnos(fichaFinalAlumnos);

    // Assert
    const notasFinales: CalificacionesAlumno[] = [
      {
        alumno: "Juan Pérez",
        notaFinal: 7.7,
      },
      {
        alumno: "María López",
        notaFinal: 3.7,
      },
    ];

    expect(result).toEqual(notasFinales);
  });
});

```

Y por último vamos a implementar el método `calcularAlumnosNotaFinal`:

Pistas:

- Usar el método `map` para recorrer el array de `calcularCalificacionAlumnos`.
- Usar el método `calcularNotaMedia` para calcular la nota media de los ejercicios prácticos y la nota media de los exámenes.
- Usar el método `calcularNotaFinal` para calcular la nota final.

Sino encuentras la solución, no desesperes, aquí la tienes:

```
export const calcularCalificacionAlumnos = (
  fichaFinalAlumnos: FichaFinalAlumno[]
): CalificacionesAlumno[] => {
  const notasFinales: CalificacionesAlumno[] = fichaFinalAlumnos.map(
    (fichaFinalAlumno) => {
      const notaMediaPracticas = calcularNotaMedia(
        fichaFinalAlumno.notaPracticas
      );
      const notaMediaExamenenes = calcularNotaMedia(
        fichaFinalAlumno.notaExamenenes
      );
      const notaFinal = calcularNotaFinal(
        notaMediaPracticas,
        notaMediaExamenenes
      );

      return {
        alumno: fichaFinalAlumno.alumno,
        notaFinal,
      };
    }
  );

  return notasFinales;
};
```

Pesos configurables

Cómo vemos en la función `calcularNotaFinal`, tenemos que multiplicar la nota media de los ejercicios prácticos por 0.6 y la nota media de los exámenes por 0.4. No está mal tener esto fijo, pero lo ideal es que sea configurable, que se pueda cambiar el peso de los ejercicios prácticos y de los exámenes.

Para no modificar nada vamos a crearnos un archivo nuevo `calcular-nota-final.ts`.

Lo ideal sería montar un interfaz y tener dos parámetros,

Pero ya que estamos podemos pasarle:

- Un array de arrays de notas.
- Un array de pesos

Y ya lo tenemos genérico para cualquier asignatura, por ejemplo podrían tener notas de prácticas, exámenes y exposiciones...

```
const calcularNotaFinal = (  
  notasMedias: number[],  
  peso: number[],  
) : number => {  
  // TODO  
};
```

Parámetros de entrada

Los parámetros de entrada son dos números:

- **notasMedias**: que es un array de números con las notas medias de los ejercicios prácticos, exámenes, exposiciones...
- **peso**: que es un array de números con el peso de los ejercicios prácticos, exámenes, exposiciones...

Parámetros de salida

Como salida vamos a devolver un número con la nota final.

Vamos a implementar una batería de pruebas para el método **calcularNotaFinal**:

calcular-nota-final.spec.ts

```
describe("calcularNotaFinal", () => {  
  it("debería devolver la nota final", () => {  
    // Arrange  
    const notasMedias = [8, 7];  
    const peso = [0.6, 0.4];  
  
    // Act  
    const result = calcularNotaFinal(notasMedias, peso);  
  
    // Assert  
    const notaFinal = 7.6;  
  
    expect(result).toEqual(notaFinal);  
  });  
  
  it("debería devolver la nota final", () => {  
    // Arrange  
    const notasMedias = [6, 7];  
    const peso = [0.6, 0.4];  
  
    // Act  
    const result = calcularNotaFinal(notasMedias, peso);  
  
    // Assert
```

```
    const notaFinal = 6.4;

    expect(result).toEqual(notaFinal);
  });

it("debería devolver la nota final", () => {
  // Arrange
  const notasMedias = [3, 2];
  const peso = [0.6, 0.4];

  // Act
  const result = calcularNotaFinal(notasMedias, peso);

  // Assert
  const notaFinal = 2.6;

  expect(result).toEqual(notaFinal);
});
```

Vamos a implementar el método `calcularNotaFinal`:

Pistas:

- Para calcular la nota final, multiplicamos la nota media de los ejercicios con su peso.
- Usar el método `toFixed` para redondear la nota final a dos decimales.

Sino has encontrado la solución, no desesperes, aquí la tienes:

```
export const calcularNotaFinal = (
  notasMedias: number[],
  peso: number[]
): number => {
  const notaFinal = notasMedias.reduce(
    (acc, notaMedia, index) => acc + notaMedia * peso[index],
    0
  );
  const notaFinalRedondeada = Number(notaFinal.toFixed(2));

  return notaFinalRedondeada;
};
```