

Trabajando con Genéricos

Código

```
// Función para mostrar elementos de un array
function mostrarElementosArray<T>(array: T[]): void
{
    array.forEach((e, i) =>
    {
        console.log(`Posición ${i}:`, e);
    });
}

// Contenido del ejercicio anterior:
interface Persona
{
    nombre: string;
    apellidos: string;
    genero: string;
    telefono: string;
    direccion: Direccion;
}

interface Direccion
{
    calle: string;
    codigoPostal: string;
    localidad: string;
    provincia: string;
}

interface Jugador extends Persona
{
    categoria: number;
}

interface Arbitro extends Persona
{
    numeroColegiado: string;
}
```

```

class JugadorImpl implements Jugador
{
    constructor(
        public nombre: string,
        public apellidos: string,
        public genero: string,
        public telefono: string,
        public direccion: Direccion,
        public categoria: number
    )
    {}
}

class ArbitroImpl implements Arbitro
{
    constructor(
        public nombre: string,
        public apellidos: string,
        public genero: string,
        public telefono: string,
        public direccion: Direccion,
        public numeroColegiado: string
    )
    {}
}

class Equipo
{
    constructor(public jugador1: Jugador, public jugador2: Jugador)
    {}

    getNombreEquipo(): string
    {
        return `${this.jugador1.apellidos}-${this.jugador2.apellidos}`;
    }
}

class Set
{
    constructor(public juegosEquipo1: number, public juegosEquipo2: number)
    {}
}

```

```

class Partido
{
    private sets: Set[] = [];

    constructor(
        public equipo1: Equipo,
        public equipo2: Equipo,
        public fecha: Date,
        public recinto: string,
        public pista: string,
        public arbitro: Arbitro
    )
    {}

    agregarSet(juegosEquipo1: number, juegosEquipo2: number): void
    {
        this.sets.push(new Set(juegosEquipo1, juegosEquipo2));
    }

    obtenerResultado(): string
    {
        let setsGanadosEquipo1 = 0;
        let setsGanadosEquipo2 = 0;

        for (const set of this.sets)
        {
            if (set.juegosEquipo1 > set.juegosEquipo2)
            {
                setsGanadosEquipo1++;
            }
            else if (set.juegosEquipo2 > set.juegosEquipo1)
            {
                setsGanadosEquipo2++;
            }
        }

        const equipoGanador = setsGanadosEquipo1 > setsGanadosEquipo2 ?
this.equipo1 : this.equipo2;
        const resultado = `${equipoGanador.getNombreEquipo()} ha ganado por
${setsGanadosEquipo1} sets a ${setsGanadosEquipo2}.`;
        return resultado;
    }
}

```

```

class Calendario
{
    private partidos: Partido[] = [];

    agregarPartido(partido: Partido): void
    {
        this.partidos.push(partido);
    }

    obtenerPartidos(): Partido[]
    {
        return this.partidos;
    }
}

const direccion1: Direccion = { calle: "Calle 1", codigoPostal: "12345",
localidad: "Ciudad A", provincia: "Provincia A" };
const direccion2: Direccion = { calle: "Calle 2", codigoPostal: "67890",
localidad: "Ciudad B", provincia: "Provincia B" };

const jugador1 = new JugadorImpl("Juan", "López", "M", "111-111-111",
direccion1, 3);
const jugador2 = new JugadorImpl("Pedro", "Martín", "M", "222-222-222",
direccion2, 4);
const jugador3 = new JugadorImpl("Luis", "Carrasco", "M", "333-333-333",
direccion1, 5);
const jugador4 = new JugadorImpl("Carlos", "Beltrán", "M", "444-444-444",
direccion2, 2);
const jugador5 = new JugadorImpl("Pepe", "García", "M", "555-555-555",
direccion1, 1);
const jugador6 = new JugadorImpl("Teresa", "Murillo", "F", "666-666-666",
direccion2, 6);

const equipo1 = new Equipo(jugador1, jugador2);
const equipo2 = new Equipo(jugador3, jugador4);
const equipo3 = new Equipo(jugador5, jugador6);

const arbitro1 = new ArbitroImpl("Ana", "Gómez", "F", "777-777-777",
direccion1, "ARB001");
const arbitro2 = new ArbitroImpl("María", "Pérez", "F", "888-888-888",
direccion2, "ARB002");

const calendario = new Calendario();

const partido1 = new Partido(equipo1, equipo2, new Date("2023-10-
01T10:00:00"), "Polideportivo A", "Pista 1", arbitro1);

```

```

partido1.agregarSet(6, 4);
partido1.agregarSet(3, 6);
partido1.agregarSet(5, 7);

const partido2 = new Partido(equipo2, equipo3, new Date("2023-10-05T10:00:00"), "Polideportivo B", "Pista 2", arbitro2);
partido2.agregarSet(3, 6);
partido2.agregarSet(4, 6);

const partido3 = new Partido(equipo1, equipo3, new Date("2023-10-08T10:00:00"), "Polideportivo A", "Pista 1", arbitro1);
partido3.agregarSet(6, 7);
partido3.agregarSet(7, 5);
partido3.agregarSet(6, 3);

calendario.agregarPartido(partido1);
calendario.agregarPartido(partido2);
calendario.agregarPartido(partido3);

// Array con todos los jugadores
const jugadoresCampeonato = [jugador1, jugador2, jugador3, jugador4, jugador5, jugador6];

// Mostramos la lista del calendario
console.log("Calendario de partidos:");
mostrarElementosArray(calendario.obtenerPartidos());

// Mostramos la lista de jugadores
console.log("Jugadores del campeonato:");
mostrarElementosArray(jugadoresCampeonato);

```

Resultado

```

Calendario de Partidos:
Posición 0: ▶ Partido {sets: Array(3), equipo1: Equipo, equipo2: Equipo, fecha: Sun Oct 01 2023 10:00:00 GMT+0200 (hora de verano de Europa central), recinto: 'Polideportivo A', ...} di313.ts:6
Posición 1: ▶ Partido {sets: Array(2), equipo1: Equipo, equipo2: Equipo, fecha: Thu Oct 05 2023 10:00:00 GMT+0200 (hora de verano de Europa central), recinto: 'Polideportivo B', ...} di313.ts:6
Posición 2: ▶ Partido {sets: Array(3), equipo1: Equipo, equipo2: Equipo, fecha: Sun Oct 08 2023 10:00:00 GMT+0200 (hora de verano de Europa central), recinto: 'Polideportivo A', ...} di313.ts:6
Jugadores del Campeonato:
Posición 0: ▶ JugadorImpl {nombre: 'Juan', apellidos: 'López', genero: 'M', telefono: '111-111-111', direccion: {...}, ...} di313.ts:6
Posición 1: ▶ JugadorImpl {nombre: 'Pedro', apellidos: 'Martín', genero: 'M', telefono: '222-222-222', direccion: {...}, ...} di313.ts:6
Posición 2: ▶ JugadorImpl {nombre: 'Luis', apellidos: 'Carrasco', genero: 'M', telefono: '333-333-333', direccion: {...}, ...} di313.ts:6
Posición 3: ▶ JugadorImpl {nombre: 'Carlos', apellidos: 'Beltrán', genero: 'M', telefono: '444-444-444', direccion: {...}, ...} di313.ts:6
Posición 4: ▶ JugadorImpl {nombre: 'Pepé', apellidos: 'García', genero: 'M', telefono: '555-555-555', direccion: {...}, ...} di313.ts:6
Posición 5: ▶ JugadorImpl {nombre: 'Teresa', apellidos: 'Murillo', genero: 'F', telefono: '666-666-666', direccion: {...}, ...} di313.ts:6

```