

PI/PJS - Samostalni zadatak za vježbu 5

```
//1.
function Osoba(ime, prezime, godina_rodenja, spol, visina) {
    this.ime = ime;
    this.prezime = prezime;
    this.godina_rodenja = godina_rodenja;
    this.spol = spol;
    this.visina = visina;
    this.predstaviSe = function() {
        console.log("Ime: " + this.ime + ", Prezime: " + this.prezime + ", Godina rođenja: " + this.godina_rodenja + ", Spol: " + this.spol + ", Visina: " + this.visina + " cm");
    };
}

let osobe = [];
osobe.push(new Osoba("Marko", "Marković", 1990, "muški", 182));
osobe.push(new Osoba("Ana", "Anić", 1985, "ženski", 168));
osobe.push(new Osoba("Ivan", "Ivić", 1992, "muški", 175));

for (let osoba of osobe) {
    osoba.predstaviSe();
}

function prosjecnaVisina(osobe) {
    var ukupnaVisina = 0;

    for (let osoba of osobe) {
        ukupnaVisina += osoba.visina;
    }

    var prosjek = ukupnaVisina / osobe.length;
    console.log("Prosjecna visina je " + prosjek.toFixed(2) + " cm");
}

prosjecnaVisina(osobe);
```

```
//2.
function Sportas(ime, prezime, godina_rodenja, spol, visina, tezina, sport, klub, broj_dresa) {
    this.ime = ime;
    this.prezime = prezime;
    this.godina_rodenja = godina_rodenja;
    this.spol = spol;
    this.visina = visina;
    this.tezina = tezina;
    this.sport = sport;
    this.klub = klub;
    this.broj_dresa = broj_dresa;
}
```

```

    this.broj_dresa = broj_dresa;
    this.nastupi = [];

    this.dodajNastup = function(nastup) {
        this.nastupi.push(nastup);
    };

    this.nastupiSportasa = function() {
        let output = '';
        for (let i = 0; i < this.nastupi.length; i++) {
            output += `Nastup ${i + 1}: ${this.nastupi[i]}, `;
        }
        console.log(output.slice(0, -2));
    };

    this.dohvatiZadnjaDvaNastupa = function() {
        let zadnjaDva = this.nastupi.slice(-2);
        console.log(`Zadnji nastupi: ${zadnjaDva.join(', ')}`);
    };
}

let sportasi = [];
sportasi.push(new Sportas("Mario", "Maric", 1985, "M", 190, 85, "Nogomet", "Dinamo", 10));
sportasi.push(new Sportas("Ana", "Anic", 1990, "Ž", 170, 60, "Tenis", "Mladost", 12));
sportasi.push(new Sportas("Ivan", "Ivic", 1995, "M", 185, 92, "Košarka", "Cibona", 23));

sportasi[0].dodajNastup("2022 Zagreb Open");
sportasi[1].dodajNastup("2023 US Open");
sportasi[2].dodajNastup("2022 NBA Playoff");

function prosjecnaTezina(sportasi) {
    let totalTezina = 0;
    let count = 0;
    for (const sportas of sportasi) {
        totalTezina += sportas.tezina;
        count++;
    }
    console.log(`Prosjecna tezina: ${totalTezina / count}`);
}

function najteziSportas(sportasi) {
    let najtezi = sportasi[0];
    for (const sportas of sportasi) {
        if (sportas.tezina > najtezi.tezina) {
            najtezi = sportas;
        }
    }
    return najtezi;
}

let sportasi_senior = [];
for (const sportas of sportasi) {

```

```

    let godina = new Date().getFullYear();
    if (godina - sportas.godina_rodenja > 30) {
        sportasi_senior.push(sportas);
    }
}

function izbrisiSvimaPrviNastup(sportasi) {
    for (const sportas of sportasi) {
        sportas.nastupi.shift();
    }
}

function azurirajBrojDresa(sportasi, brojDresa, noviBrojDresa) {
    let sportas = sportasi.find(function(s) {
        return s.broj_dresa === brojDresa;
    });

    if (!sportas) {
        console.log('Sportaš s tim brojem dresa nije pronađen.');
```

```

//3.
function gcd_array(array) {
    // Počinjemo s GCD prvog broja u nizu, koji je sam po sebi
    let result = array[0];

    // Prolazimo kroz sve ostale brojeve u nizu
    for (let i = 1; i < array.length; i++) {
        // Računamo GCD trenutnog rezultata i trenutnog broja u nizu
        result = gcd_two_numbers(result, array[i]);

        // Ako je GCD postao 1, možemo prekinuti jer je GCD cijelog niza 1
        if (result === 1) {
            return 1;
        }
    }
    // Vraćamo konačni GCD svih brojeva u nizu
    return result;
}

```