

Tartalom

A függvényekről TypeScript-ben	2
Visszatérési érték, és paraméter nélküli eljárások.....	2
Visszatérési érték nélküli, paraméteres eljárások.....	2
Visszatérési értékes, paraméter nélküli függvények.....	2
Visszatérési értékes, paraméteres függvények.....	3
Visszatérési értékes, paraméteres függvények, TUPLE adatszerkezettel	3
Tuple függvény eredményének meghívása:.....	3

A függvényekről TypeScript-ben

A függvények hasonlóan működnek, mint JavaScriptben, lévén arra fordul le a végén, a tsc parancs kiadása után, de ugyanakkor, a gyengén típusosság, erősen típusosságra vált át, ami azt jelenti, minden egyes változót, amit használunk, és annak típusát, meg kell adnunk!

Visszatérési érték, és paraméter nélküli eljárások

Ebben az esetben nagyon minimális az eltérés, de azért van, ugyanis azokat a függvényeket amiknek nincs visszatérési értékük, fel kell ruházni, a void tulajdonsággal a következő módon.

```
function IloveTypeScript(): void {  
    document.write("Szeretem a TypeScriptet");  
}
```

Visszatérési érték nélküli, paraméteres eljárások

Amennyiben paramétereket szeretnénk használni, meg kell adnunk azok neve mellett szintén, azok típusát is, ahogy korábban a primitív változó típusokkal tettük ezt.

```
function NegyzetKeruletTeruletEljaras(a: number): void {  
    let kerulet = 4 * a;  
    let terület = a * a;  
    document.write(`A ${a} oldalú négyzet<br>  
kerülete:${kerulet}<br>területe:${terület}`);  
}
```

Visszatérési értékes, paraméter nélküli függvények

Ha csak simán visszatérési értékes függvényt szeretnénk, akkor a függvény neve utáni kerek zárójeleket követően, a void helyett meg kell adni annak típusát, például.: number, string, boolean, vagy épp egyéb összetett adatszerkezet

```
function RandomGeneralo100Eljaras(): number {  
    return Math.round(Math.random() * 100);  
}
```

Visszatérési értékes, paraméteres függvények

Végül, ha visszatérési értékes és paraméteres is egy függvény, akkor szükségünk van a paraméterek típusának megadására és a visszatérési értéket is megadni!

```
function NegyzetKeruletFuggveny(a: number): number {  
    return 4 * a;  
}
```

Visszatérési értékes, paraméteres függvények, TUPLE adatszerkezettel

Előfordulhat, hogy szeretnénk több értékkel is visszatérni, ezeket az úgynevezett Tuple adatszerkezettel adhatjuk meg, ilyenkor a visszatérési érték, egy tömbként viselkedik, melyben a tuple-ban lévő elemekre, az adott változó „indexeként” hivatkozhatunk!

```
function NegyzetKeruletTupleFuggveny(a: number): [number, number] {  
    let keruletEredmeny = 4 * a;  
    let területEredmeny = a * a;  
    return [keruletEredmeny, területEredmeny];  
}
```

Tuple függvény eredményének meghívása:

```
document.write("<br>Tuple értékkel való függvény eredménye:")  
var tupleFuggvenyEredmeny = NegyzetKeruletTupleFuggveny(5);  
document.write("<br>A négyzet kerülete:" + tupleFuggvenyEredmeny[0]);  
document.write("<br>A négyzet kerülete:" + tupleFuggvenyEredmeny[1]);
```