

## آموزش TypeScript – آموزش اتحاد (Union) در تایپ اسکریپت (TypeScript)

در نسخه ی 1.4 تایپ اسکریپت این امکان برای برنامه ها فراهم شده است تا یک یا دو نوع را با یکدیگر ترکیب کنند. نوع یونیون روشی قدرتمند برای بیان و ابراز مقداری است که این مقدار می تواند جزئی از چندین نوع داده باشد. دو یا چند نوع داده ای با استفاده از علامت (|) با یکدیگر ترکیب می شوند و نوع یونیون را نشان می دهند. به بیان دیگر، نوع یونیون به صورت ترتیبی از نوع های جدا شده توسط خط های عمودی، نوشته می شود.

سینتکس : لفظ یونیون

```
Type1|Type2|Type3
```

مثال متغیر نوع یونیون

```
var val:string|number  
  
val = 12  
  
console.log("numeric value of val "+val)  
  
val = "This is a string"  
  
console.log("string value of val "+val)
```

نوع متغیر در مثال بالا یونیون است. این یعنی این متغیر می تواند به عنوان مقدار خود، شامل عدد و یا شامل رشته باشد.

بعد از کامپایل کردن کد بالا، کد جاوا اسکریپت زیر ایجاد می شود.

```
//Generated by typescript 1.8.10  
  
var val;  
  
val = 12;  
  
console.log("numeric value of val " + val);  
  
val = "This is a string";  
  
console.log("string value of val " + val);
```

و خروجی به صورت زیر نمایش داده می شود.

```
numeric value of val 12
```

```
string value of val this is a string
```

مثال : نوع یونیون و پارامتر تابع

```
function disp(name:string|string[]) {  
  
    if(typeof name == "string") {  
  
        console.log(name)  
  
    } else {  
  
        var i;  
  
        for(i = 0;i<name.length;i++) {  
  
            console.log(name[i])  
  
        }  
  
    }  
  
}  
  
disp("mark")  
  
console.log("Printing names array....")  
  
disp(["Mark","Tom","Mary","John"])
```

تابع disp() می تواند آرگومانی را بپذیرد که نوع آن می تواند هم رشته و هم آرایه ی رشته ای باشد.

بعد از کامپایل کردن کد بالا، کد جاوا اسکریپت زیر ایجاد می شود.

```
//Generated by typescript 1.8.10
```

```
function disp(name) {  
  
    if (typeof name == "string") {  
  
        console.log(name);  
  
    } else {  
  
        var i;
```

```
for (i = 0; i < name.length; i++) {  
    console.log(name[i]);  
}  
  
}  
  
}  
  
disp("mark");  
  
console.log("Printing names array....");  
  
disp(["Mark", "Tom", "Mary", "John"]);
```

و خروجی به صورت زیر نمایش داده می شود.

```
Mark  
Printing names array....  
Mark  
Tom  
Mary  
John
```

## آموزش انواع یونیون و آرایه ها در TypeScript

نوع یونیون را می توان در آرایه ها، property ها و رابط ها، به کار برد. در مثال زیر کاربرد نوع یونیون با یک آرایه نشان داده شده است.

مثال: نوع یونیون و آرایه

```
var arr:number[]|string[];  
  
var i:number;  
  
arr = [1,2,4]  
  
console.log("***numeric array**")  
  
for(i = 0;i<arr.length;i++) {
```

```
console.log(arr[i])
}

arr = ["Mumbai", "Pune", "Delhi"]
console.log("**string array**")

for(i = 0; i < arr.length; i++) {
  console.log(arr[i])
}
```

در این برنامه آرایه ای اعلان شده است. این آرایه می تواند بیانگر مجموعه ای عددی و یا مجموعه ای رشته ای باشد.

بعد از کامپایل کردن کد بالا، کد جاوا اسکریپت زیر ایجاد می شود.

```
//Generated by typescript 1.8.10

var arr;

var i;

arr = [1, 2, 4];

console.log("**numeric array**");

for (i = 0; i < arr.length; i++) {
  console.log(arr[i]);
}

arr = ["Mumbai", "Pune", "Delhi"];

console.log("**string array**");
```

```
for (i = 0; i < arr.length; i++) {  
  
  console.log(arr[i]);  
  
}
```

و خروجی به صورت زیر نمایش داده می شود.

```
**numeric array**  
1  
2  
4  
  
**string array**  
Mumbai  
Pune  
Delhi
```