TypeScript

Tahaluf Training Center 2021

شركة تحالف الإمارات للحلول التقنية ذ.م.م.
TAHALUF AL EMARAT TECHNICAL SOLUTIONS L.L.C.

Day 02

1. If Condition
2. Switch Statements
3. Loops (For, While)
4. Object
5. Array
6. Arrow Function

IF Condition

- An if statement can include one or more expressions which return boolean. If the boolean expression evaluates to true, a set of statements is then executed.

Example:

```
if (true)
{
    console.log('This will always executed if the
is true.');
}

if (false) {
    console.log('This will never executed becoue
the condition is false.');
}
```

IF Condition

- In this example, the if condition expression num1 < num2 is evaluated to true and so it executes the statement within the curly { } brackets.

Example :

```
let num1: number = 10, num2 = 20;

if (x < y)
{
  console.log('num1 is less than num2');
}
```

IF Condition

- An if else condition incorporates two squares - if block and an else block. Assuming the if condition assesses to the true, the in case block is executed. Something else, the else block is executed .

Example:

```
let num1: number = 10, num1 = 20;
if (num1 > num2)
{ console.log('num1 is greater than num2.');
}

else
{ console.log('num1 is less than or equal to num2.')
; //This statement will never be executed

}
```

Else if condition

```
let num1: number = 10, num2 = 20;
if (num1 > num2)
{
    console.log('num1 is greater than num2.');
}
else if (num1 < num2)
{
    console.log('num1 is less than num2.'); //This will be executed
}
else if (num1 == num2)
{
    console.log('num1 is equal to num2');
}
```

شركة تحالف الإمارات للحلول التقنية ذ.م.م.
TAHALUF AL EMARAT TECHNICAL SOLUTIONS L.L.C.

Day 02

1. If Condition
2. **Switch Statements**
3. Loops (For, While)
4. Object
5. Array
6. Arrow Function

- **A switch** statement has one block of code corresponding to each value and can have any number of such blocks. When the match to a value is found, the corresponding block of code is executed.

Example:

```
switch(expression) {
   case constant-expression1: {
      //statements;
      break;
   }
   case constant_expression2: {
      //statements;
      break;
   }
   default: {
      //statements;
      break;
   } }
```

Switch statement

- The switch statement can include constant or variable expression which can return a value of any data type.
- There can be any number of case statements within a switch. The case can include a constant or an expression.

Switch statement

- We must use break keyword at the end of each case block to stop the execution of the case block.
- The return type of the switch expression and case expression must match.
- The default block is optional.

Demo

TS FirstProg.ts > ...

```typescript
let day : number = 4;

switch (day) {
    case 0:
        console.log("It is a Sunday.");
        break;
    case 1:
        console.log("It is a Monday.");
        break;
    case 2:
        console.log("It is a Tuesday.");
        break;
    case 3:
        console.log("It is a Wednesday.");
        break;
    case 4:
        console.log("It is a Thursday.");
        break;
    case 5:
        console.log("It is a Friday.");
        break;
    case 6:
        console.log("It is a Saturday.");
        break;
    default:
        console.log("No such day exists!");
        break;
}
```

شركة تحالف الإمارات للحلول التقنية ذ.م.م.
TAHALUF AL EMARAT TECHNICAL SOLUTIONS L.L.C.

Day 02

1. If Condition
2. Switch Statements
3. **Loops (For, While)**
4. Object
5. Array
6. Arrow Function

For Loop

- The for loop is used to execute a block of code a given number of times, which is specified by a condition.

For Loop

Example:

```
for (let i = 0; i < 3; i++) {
  console.log ("Block statement execution no." + i);
}
```

Output:

Block statement execution no.0
Block statement execution no.1
Block statement execution no.2

For Loop

شركة تحالف الإمارات للحلول التقنية ذ.م.م.
TAHALUF AL EMARAT TECHNICAL SOLUTIONS L.L.C.

TS FirstProg.ts > ...

```ts
let arr = [10, 20, 30, 40];

for (var val of arr) {
  console.log(val); // prints values: 10, 20, 30, 40
}
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\User\Desktop\TypeScript\Day01\Demo> tsc .\FirstProg.ts
PS C:\Users\User\Desktop\TypeScript\Day01\Demo> node .\Firstprog.js
10
20
30
40
PS C:\Users\User\Desktop\TypeScript\Day01\Demo>
```

While Loop

- The while loop is another type of loop that checks for a specified condition before beginning to execute the block of statements. The loop runs until the condition value is met.

Example:

```
let i: number = 2;
while (i < 4) {
    console.log( "Block statement execution no." +
i )
    i++; }
```

do. While loop

- The do..while loop is similar to the while loop, except that the condition is given at the end of the loop. The do..while loop runs the block of code at least once before checking for the specified condition. For the rest of the iterations, it runs the block of code only if the specified condition is met.

Example:

```
let i: number = 2;
do {
    console.log("Block statement execution no." + i )
    i++;
} while ( i < 4)
```

شركة تحالف الإمارات للحلول التقنية ذ.م.م.
TAHALUF AL EMARAT TECHNICAL SOLUTIONS L.L.C.

Day 02

1. If Condition
2. Switch Statements
3. Loops (For, While)
4. **Object**
5. Array
6. Arrow Function

Object

An **object** is an instance which contains set of key value pairs.

Syntax:

```
var object_name = {
   key1: “value1”, //scalar value
   key2: “value”,
   key3: function() {
      //functions
   },
   key4:[“content1”, “content2”] //collection
};
```

❖ **Example :**

```
const employee: any = {
            name: 'Ahmad',
            age: 30,
            department: 'web Development',
            section: 1
        }
```

- You can edit to the constant object using the name of the properties .

- Example :

```
employee.name = 'Dana’;
```

- And you can add a new properties for a constant object

- Example :
- `employee.address = 'Jordan/Irbid '`

شركة تحالف الإمارات للحلول التقنية ذ.م.م.
TAHALUF AL EMARAT TECHNICAL SOLUTIONS L.L.C.

Day 02

1. If Condition
2. Switch Statements
3. Loops (For, While)
4. Object
5. **Array**
6. Arrow Function

Array

- An array is a special type of data type which can store multiple values of different data types sequentially.

- Syntax :

```
const , var, let Array_Name: DataType[] =[values]
```

- To add to the constant array you can use push method .

- **Example**

```
const nameArray: string[] = ['name1’,
'name2', 'name3', 'name4']
    nameArray.push('name5');
    nameArray.push('name6');
    Filevalidation = () => {
```

- To change a specific value for constant array :
- Syntax :

```
Array_name[Index] = newValue;
```

- Example :

```
nameArray[0] = 'name6';
```

- There are two ways to declare an array:

1. Using square brackets.
   This method is similar to how you would declare arrays in JavaScript.

```
let names: string[] = [‘Dana', ‘Kanaan’];
```

2. Using a generic array type, Array<elementType>.

```
let names: Array<string> = [‘Dana', ‘Kanaan'];
```

spread operator:

The main objective of the spread operator is to *spread* the elements of an array or object. This is best explained with examples.

```
const sport: string[] = ['Basket ball',
'Tennis', 'Football', 'soccer']
                const fightingSports:
string[] = ['MMA', 'Kung Fu'];
                const allSports =
[...sport, ...fightingSports];
                console.log(allSports);
```

You can add to spread operator:

```
const sport: string[] = ['Basket ball',
'Tennis', 'Football', 'soccer']
                const fightingSports:
string[] = ['MMA', 'Kung Fu'];
                const allSports =
['AnotherSport3', ...sport,
'AnotherSport4', ...fightingSports,
'AnotherSport5'];
                console.log(allSports);
```

Distructuring : Swap two variables without using a third one

- Note that array destructuring is effectively the compiler doing the [0], [1], ...

Example:

```
 const sport: string[] = ['Basket ball',
'Tennis', 'Football', 'soccer']
const [top1, top2] = sport;
console.log(top1, top2)//Basket ball,Tennis
```

Example:

```
 const sport: string[] = ['Basket ball',
'Tennis', 'Football', 'soccer']
                const [top1, top2,
...otherSports] = sport;
                console.log(top1, top2);
                console.log(otherSports);
```

Sreach about Distructuring in the object

شركة تحالف الإمارات للحلول التقنية ذ.م.م.
TAHALUF AL EMARAT TECHNICAL SOLUTIONS L.L.C.

Day 02

1. If Condition
2. Switch Statements
3. Loops (For, While)
4. Object
5. Array
6. **Arrow Function**

❖ **Syntax**

```
const FunctionName =
(Parameters) => {//Body of the function};
```

❖ **Example :**

```
const addNumbers =
(num1: number, num2: number) => num1 + num2;
        const sum = addNumbers(5, 6);
        console.log(sum);
```

Arrow Function

```
 type MyFunctionTypes = (() => void) |
string
                const getMyPrinter =
(printType: string, n1: string, n2: string):
MyFunctionTypes => {
                    if (printType ==
'fullName') {
                        return () => {
                            return
console.log(`${n1} ${n2}`)

                        };
                    }
```

```
 else if (printType == 'comma') {
                        return () => {
return console.log(`${n1}, ${n2}`) }
                    }
                    else {
                        return n1 + n2;
                    }
                }
```

Arrow Function

```
const printFun = getMyPrinter('fullName',
'Angular', 'Typescript');
                const printFun2 =
getMyPrinter('comma', 'Angular',
'Typescript');
                const printFun3 =
getMyPrinter('SomethingElse', 'Angular',
'Typescript');
```

Arrow Function

```
const printFormPrinter = (myPrinter:
MyFunctionTypes) => (typeof myPrinter ==
'string') ?

console.log(myPrinter) : myPrinter();

printFormPrinter(printFun);

printFormPrinter(printFun2);

printFormPrinter(printFun3);
```