

JavaScript Scope (Πεδίο Εκτέλεσης)

Το **scope** στη JavaScript καθορίζει από πού μπορείς να έχεις πρόσβαση σε μεταβλητές, συναρτήσεις και αντικείμενα στον κώδικά σου. Υπάρχουν τέσσερα βασικά είδη scope:

1. **Global Scope (Καθολικό Πεδίο)**
2. **Function Scope (Πεδίο Συνάρτησης)**
3. **Block Scope (Πεδίο Μπλοκ)**
4. **Lexical Scope (Λεξικό Πεδίο)**

Ας τα εξετάσουμε ένα προς ένα με δομημένα παραδείγματα.

1. Global Scope (Καθολικό Πεδίο)

Το **global scope** περιλαμβάνει μεταβλητές που δηλώνονται εκτός οποιασδήποτε συνάρτησης ή μπλοκ. Οι μεταβλητές αυτές είναι προσβάσιμες παντού στον κώδικα, μετά την εκτέλεσή τους.

```
javascript
Copy code
let globalVariable = "I am global";

function test() {
  console.log(globalVariable); // Εκτυπώνει: "I am global"
}

test();
console.log(globalVariable); // Εκτυπώνει: "I am global"
```

- Μεταβλητές στο **global scope** είναι ορατές και προσβάσιμες από οποιοδήποτε μέρος του κώδικα.

2. Function Scope (Πεδίο Συνάρτησης)

Όταν δηλώνεις μεταβλητές μέσα σε μια συνάρτηση, αυτές βρίσκονται σε **function scope**. Οι μεταβλητές αυτές είναι προσβάσιμες μόνο μέσα στη

συνάρτηση όπου δηλώθηκαν.

```
javascript
Copy code
function myFunction() {
  let localVariable = "I am local";
  console.log(localVariable); // Εκτυπώνει: "I am local"
}

myFunction();
console.log(localVariable); // Σφάλμα: localVariable is not
defined
```

- Μεταβλητές σε **function scope** δεν είναι ορατές έξω από τη συνάρτηση όπου δηλώθηκαν.

3. Block Scope (Πεδίο Μπλοκ)

Το **block scope** αναφέρεται στις μεταβλητές που δηλώνονται μέσα σε μπλοκ κώδικα, όπως `if`, `for`, `while` κ.λπ., όταν χρησιμοποιούνται οι δηλωτικοί `let` ή `const`.

```
javascript
Copy code
if (true) {
  let blockVariable = "I am block scoped";
  console.log(blockVariable); // Εκτυπώνει: "I am block sco
ped"
}

console.log(blockVariable); // Σφάλμα: blockVariable is not
defined
```

- Οι μεταβλητές στο **block scope** είναι ορατές μόνο μέσα στο μπλοκ όπου δηλώθηκαν.

4. Lexical Scope (Λεξικό Πεδίο)

Το **lexical scope** ορίζει ότι οι εσωτερικές συναρτήσεις έχουν πρόσβαση στις μεταβλητές των εξωτερικών συναρτήσεων, ενώ το αντίστροφο δεν ισχύει. Το λεξικό πεδίο "βλέπει" τη δομή του κώδικα κατά τη δήλωση, όχι κατά την εκτέλεση.

```
javascript
Copy code
function outerFunction() {
  let outerVariable = "I am from outer";

  function innerFunction() {
    console.log(outerVariable); // Εκτυπώνει: "I am from ou
ter"
  }

  innerFunction();
}

outerFunction();
```

- Οι εσωτερικές συναρτήσεις μπορούν να "δουν" και να χρησιμοποιήσουν τις μεταβλητές των εξωτερικών συναρτήσεων χάρη στο **lexical scope**.