

# Guida alle Strutture di Controllo in JavaScript

## 1. L'istruzione `if`

L'istruzione `if` permette di eseguire blocchi di codice solo se una condizione è vera. Se la condizione è falsa, il codice all'interno delle parentesi graffe non verrà eseguito.

**Esempio:**

```
let x = 5;
if (x > 0) {
  console.log('x è un numero positivo!');
}
```

---

## 2. Operatori di Confronto

Gli operatori di confronto servono per confrontare due valori e restituire `true` o `false`.

- **== (Uguaglianza non stretta):** Controlla se due valori sono uguali, senza considerare il tipo.

```
28 == '28' // true
```

- **=== (Uguaglianza stretta):** Confronta anche il tipo di dato.

```
28 === '28' // false
```

- **!= e !== (Diverso e Diverso stretto):** Controllano se due valori (e tipi con `!==`) sono diversi.

```
28 !== '28' // true
```

- **Altri operatori:**
    - `>` (maggiore di), `<` (minore di)
    - `>=` (maggiore o uguale), `<=` (minore o uguale)
-

### 3. Operatori Logici

Gli operatori logici vengono usati per combinare più condizioni:

- **&& (AND):** Entrambe le condizioni devono essere vere
- **|| (OR):** Almeno una condizione deve essere vera
- **! (NOT):** Inverte il valore di verità

**Esempio:**

```
let myAge = 28;
if ((myAge >= 0 && myAge < 3) || myAge > 90) {
  console.log("Non sei al tuo massimo fisico.");
}
```

---

### 4. Truthy vs Falsy

JavaScript considera alcuni valori come `true` (truthy) e altri come `false` (falsy) senza bisogno di confronto.

- **FALSY:** `false`, `""` (stringa vuota), `0`, `-0`, `undefined`, `null`, `NaN`
- **Tutti gli altri valori sono truthy.**

**Esempio:**

```
let nome = "";
if (nome) {
  console.log("Ciao, " + nome);
} else {
  console.log("Nessun nome inserito");
}
```

---

### 5. Valutazione con "Short-Circuit"

JavaScript esegue gli operatori logici da sinistra a destra e si ferma appena trova il risultato.

- `(falsey && qualsiasi cosa) => falsey`
- `(truthy || qualsiasi cosa) => truthy`

## Esempio:

```
let numeratore = 5;
let denominatore = 0;
if (denominatore !== 0 && (numeratore / denominatore > 0)) {
  console.log("Frazione valida");
}
```

---

## 6. Istruzioni `if/else` e `if/else if/else`

Le istruzioni `if/else` permettono di eseguire un blocco di codice in base a diverse condizioni.

### Esempio semplice con `if/else`:

```
let età = 28;
if (età > 16) {
  console.log("Puoi guidare!");
} else {
  console.log("Devi aspettare " + (16 - età) + " anni.");
}
```

### Esempio con `if/else if/else`:

```
let età = 20;
if (età >= 35) {
  console.log("Puoi votare e candidarti per qualsiasi carica!");
} else if (età >= 25) {
  console.log("Puoi votare e candidarti per il Senato!");
} else if (età >= 18) {
  console.log("Puoi votare!");
} else {
  console.log("Non puoi votare.");
}
```

---

## 7. Cicli ( `while` e `for` )

I cicli permettono di eseguire ripetutamente un blocco di codice finché una condizione è vera.

- Ciclo `while` (ripete il codice finché la condizione è vera):

```
let x = 0;
while (x < 5) {
  console.log(x);
  x++;
}
```

- **Ciclo for** (ha una struttura più compatta):

```
for (let i = 0; i < 5; i++) {
  console.log(i);
}
```

---

## 8. break : Uscire da un ciclo

Il comando `break` permette di interrompere un ciclo prima che termini normalmente.

**Esempio:**

```
for (let current = 100; current < 200; current++) {
  console.log("Testing " + current);
  if (current % 7 == 0) {
    console.log("Trovato! " + current);
    break;
  }
}
```

---

## 9. switch (Alternativa a if/else if )

L'istruzione `switch` è utile quando si hanno molte condizioni da controllare su una stessa variabile.

**Esempio:**

```
let giorno = 'Monday';
let messaggio;

switch (giorno) {
```

```
case 'Monday':  
    messaggio = "È lunedì!";  
    break;  
case 'Wednesday':  
    messaggio = "È mercoledì!";  
    break;  
case 'Friday':  
    messaggio = "È venerdì!";  
    break;  
default:  
    messaggio = "Giorno non valido!";  
}  
console.log(messaggio);
```

---

## 10. Switch con Fall-Through

Se si omette `break`, JavaScript continua ad eseguire i case successivi.

**Esempio:**

```
let day = 'Monday';  
let message;  
  
switch (day) {  
    case 'Monday':  
    case 'Tuesday':  
    case 'Friday':  
        message = "È un giorno lavorativo";  
        break;  
    case 'Saturday':  
    case 'Sunday':  
        message = "È il weekend!";  
        break;  
    default:  
        message = "Giorno non valido";  
}  
console.log(message);
```