

# Learn-JS-Array

## JavaScript Array - সম্পূর্ণ বিশ্লেষণ (বাংলায়)

### Part 1: Array কি এবং কিভাবে বানাতে হয় (array.js)

javascript

```
const numbers1 = [12, 22, 54, 65];  
const friends = ['Abul', 'babul', 'cabul', 'dabul', 'bulbul'];  
const passed = [true, false, true, false];  
const mixedArray = [12, 'Alahbad', true, 45.566];
```

ব্যাখ্যা:

- `numbers1` → শুধু সংখ্যা রাখা হয়েছে
- `friends` → শুধু নাম (string) রাখা হয়েছে
- `passed` → true/false (boolean) রাখা হয়েছে
- `mixedArray` → সব ধরনের জিনিস একসাথে! (number, string, boolean)

মনে রাখুন: Array তে `[]` square bracket ব্যবহার করতে হয়।

---

### Part 2: Array এর length ( কতগুলো জিনিস আছে) (length.js)

javascript

```
const numbers2 = [1, 5, 66, 11, 45, 65, 65, 88, 11, 45, 65, 65, 88, 15];  
console.log(numbers2.length); // Output: 14
```

ব্যাখ্যা:

- `.length` দিয়ে মোট কতগুলো element আছে সেটা জানা যায়
- এখানে ১৪ টা সংখ্যা আছে, তাই output হবে **14**

সহজ কথায়: Array তে কয়টা জিনিস আছে গুনতে চাইলে `.length` ব্যবহার করুন!

---

### Part 3: Index দিয়ে কাজ করা (index.js)

javascript

```
const numbers3 = [88, 77, 99, 421, 54, 63];  
  
// নির্দিষ্ট position এর value পাওয়া
```

```
console.log(numbers3[3]); // Output: 421
```

```
const fourth = numbers3[3];  
console.log(fourth); // Output: 421
```

```
// Value পরিবর্তন করা  
numbers3[1] = 999;  
console.log(numbers3); // [88, 999, 99, 421, 54, 63]
```

ব্যাখ্যা:

- Array এর position **0** থেকে শুরু হয়
  - `numbers3[0]` = 88
  - `numbers3[1]` = 77
  - `numbers3[2]` = 99
  - `numbers3[3]` = 421 ✓
- `numbers3[1] = 999` → ১ নম্বর position এ নতুন value বসালাম

মনে রাখুন: Array এর গণনা **0** থেকে শুরু!

---

## + Part 4: Push (শেষে যোগ করা) (push-pop.js)

javascript

```
const numbers4 = [12, 87, 98, 45];  
  
console.log(numbers4); // [12, 87, 98, 45]  
numbers4.push(44);  
numbers4.push(65);  
numbers4.push(98, 11, 9);  
console.log(numbers4); // [12, 87, 98, 45, 44, 65, 98, 11, 9]  
````
```

```
**ব্যখ্যা:**  
- `push()` = Array এর **শেষে** নতুন জিনিস যোগ করে  
- একসাথে একাধিক জিনিসও যোগ করা যায়: `push(98, 11, 9)`
```

```
**উদাহরণ:**  
````  
শুরু:      [12, 87, 98, 45]  
push(44):  [12, 87, 98, 45, 44]  
push(65):  [12, 87, 98, 45, 44, 65]
```

## — Part 5: Pop (শেষ থেকে সরানো) (pop.js)

javascript

---

```
const friends2 = ['balam', 'kalam', 'salam', 'gelam', 'pailam'];
console.log(friends2); // ['balam', 'kalam', 'salam', 'gelam', 'pailam']
```

```
const out1 = friends2.pop();
console.log(friends2); // ['balam', 'kalam', 'salam', 'gelam']
const out2 = friends2.pop();
console.log(friends2); // ['balam', 'kalam', 'salam']
console.log(out1, out2); // pailam gelam
````
```

**\*\*ব্যাখ্যা:\*\***

- `pop()` = Array এর **শেষের** জিনিসটা সরিয়ে দেয়  
- যে জিনিসটা সরানো হয় সেটা **return** করে

**\*\*ধাপে ধাপে:\*\***

````

শুরু: ['balam', 'kalam', 'salam', 'gelam', 'pailam']  
`pop()`: ['balam', 'kalam', 'salam', 'gelam'] → 'pailam' বের হলো  
আবার `pop()`: ['balam', 'kalam', 'salam'] → 'gelam' বের হলো

## Part 6: Shift & Unshift (শুরু থেকে কাজ করা) (shift-unshift.js)

javascript

```
const friends3 = ['balam', 'kalam', 'salam', 'gelam', 'pailam'];
console.log(friends3); // ['balam', 'kalam', 'salam', 'gelam', 'pailam']
```

```
friends3.shift(); // প্রথমটা সরানো
console.log(friends3); // ['kalam', 'salam', 'gelam', 'pailam']
```

```
friends3.unshift('ghumailam'); // শুরুতে যোগ করা
console.log(friends3); // ['ghumailam', 'kalam', 'salam', 'gelam', 'pailam']
```

**ব্যাখ্যা:**

- `shift()` = Array এর **প্রথম** জিনিসটা সরায়
- `unshift()` = Array এর **শুরুতে** নতুন জিনিস যোগ করে

**মনে রাখার কৌশল:**

- **Push/Pop** → শেষের দিকে কাজ করে
- **Shift/Unshift** → শুরুর দিকে কাজ করে

## Part 7: Includes (আছে কিনা চেক করা) (includes.js)

javascript

```
const friends4 = ['balam', 'kalam', 'salam', 'gelam', 'pailam'];

console.log(friends4.includes('gelam')); // true
```

```
console.log(friends4.includes('khailam')); // false

if(friends4.includes('salam')){
  console.log('party'); // এটা print হবে
}
else {
  console.log('no food. we are fasting')
}
```

ব্যখ্যা:

- `includes()` = কোনো জিনিস Array তে আছে কিনা check করে
- **true** return করে যদি থাকে
- **false** return করে যদি না থাকে
- if-else এর সাথে perfect কাজ করে!

## Part 8: IndexOf (কোন position এ আছে) (indexOf.js)

javascript

```
const friends5 = ['balam', 'kalam', 'salam', 'gelam', 'pailam'];

console.log(friends5.indexOf('gelam')); // 3
console.log(friends5.indexOf('tomato')); // -1
````
```

**\*\*ব্যখ্যা:\*\***

- `indexOf()` = কোন position এ আছে সেটা বলে দেয়
- Position **\*\*0\*\*** থেকে গণনা শুরু
- যদি না পায় তাহলে **\*\* -1 \*\*** return করে

**\*\*ব্রেকডাউন:\*\***

```
````
Position:  0      1      2      3      4
Value:    ['balam', 'kalam', 'salam', 'gelam', 'pailam']
           ^
           'gelam' আছে position 3 এ
```

## Part 9: IsArray (Array কিনা চেক করা) (isArray.js)

javascript

```
const friends6 = ['balam', 'kalam', 'salam', 'gelam', 'pailam'];
const nums = [];
const food = 'ros o gol l a';
const age = 15;

console.log(Array.isArray(friends6)); // true ✓
console.log(Array.isArray(nums)); // true ✓
```

```
console.log(Array.isArray(food)); // false ✗  
console.log(Array.isArray(age)); // false ✗
```

ব্যাখ্যা:

- `Array.isArray()` = কোনো জিনিস Array কিনা check করে
- Array হলে **true**, না হলে **false**
- খালি Array `[]` ও Array হিসেবে গণ্য হয়



## সারসংক্ষেপ - Quick Reference Table

| Method                       | কাজ                         | উদাহরণ                          | Result           |
|------------------------------|-----------------------------|---------------------------------|------------------|
| <code>.length</code>         | মোট element গণনা            | <code>[1,2,3].length</code>     | 3                |
| <code>[index]</code>         | নির্দিষ্ট position এর value | <code>[10,20,30][1]</code>      | 20               |
| <code>.push()</code>         | শেষে যোগ করা                | <code>arr.push(5)</code>        | শেষে 5 যোগ হবে   |
| <code>.pop()</code>          | শেষ থেকে সরানো              | <code>arr.pop()</code>          | শেষেরটা বাদ যাবে |
| <code>.shift()</code>        | শুরু থেকে সরানো             | <code>arr.shift()</code>        | প্রথমটা বাদ যাবে |
| <code>.unshift()</code>      | শুরুতে যোগ করা              | <code>arr.unshift(1)</code>     | শুরুতে 1 যোগ হবে |
| <code>.includes()</code>     | আছে কিনা                    | <code>arr.includes('x')</code>  | true/false       |
| <code>.indexOf()</code>      | কোন position এ              | <code>arr.indexOf('x')</code>   | position number  |
| <code>Array.isArray()</code> | Array কিনা                  | <code>Array.isArray(arr)</code> | true/false       |



## মনে রাখার টিপস:

1. শেষের দিকে কাজ: Push **+** এবং Pop **-**
2. শুরুর দিকে কাজ: Unshift **+** এবং Shift **-**
3. খুঁজে পেতে: includes (আছে কি নেই) এবং indexOf (কোথায় আছে)
4. Check করতে: isArray (Array কিনা)