

```
173 // 1. at(index) - Belgilangan indexdagi elementni oladi
174 const arr1 = [10, 20, 30];
175 console.log(arr1.at(1)); // Output: 20
176 // -----
177
178 // 2. concat() - Ikki massivni birlashtiradi
179 const arr2 = [1, 2];
180 const arr3 = [3, 4];
181 const combined = arr2.concat(arr3);
182 console.log(combined); // Output: [1, 2, 3, 4]
183 // -----
184
185 // 3. copyWithin() - Bir qismini boshqa joyga nusxalaydi
186 const arr4 = [1, 2, 3, 4, 5];
187 arr4.copyWithin(0, 3);
188 console.log(arr4); // Output: [4, 5, 3, 4, 5]
189 // -----
190
191 // 4. entries() - [index, value] juftliklari
192 const arr5 = ['a', 'b', 'c'];
193 for (let [i, v] of arr5.entries()) {
194   console.log(i, v);
195 }
196 // Output:
197 // 0 'a'
198 // 1 'b'
199 // 2 'c'
200 // -----
201
202 // 5. every() - Barcha elementlar shartga mos bo'lsa true
203 const arr6 = [10, 20, 30];
204 console.log(arr6.every(x => x > 5)); // Output: true
205 // -----
206
207 // 6. fill() - To'ldirish
208 const arr7 = [1, 2, 3];
209 arr7.fill(0);
210 console.log(arr7); // Output: [0, 0, 0]
211 // -----
```

```
213 // 7. filter() - Shartga mos elementlar
214 const arr8 = [5, 10, 15];
215 const filtered = arr8.filter(x => x > 5);
216 console.log(filtered); // Output: [10, 15]
217 // -----
218
219 // 8. find() - Birinchi mos element
220 console.log(arr8.find(x => x > 5)); // Output: 10
221 // -----
222
223 // 9. findIndex()
224 console.log(arr8.findIndex(x => x > 5)); // Output: 1
225 // -----
226
227 // 10. flat() - Massivni tekislash
228 const arr9 = [1, [2, [3]]];
229 console.log(arr9.flat(2)); // Output: [1, 2, 3]
230 // -----
231
232 // 11. forEach()
233 arr8.forEach(x => console.log(x));
234 // Output:
235 // 5
236 // 10
237 // 15
238 // -----
239
240 // 12. includes()
241 console.log(arr8.includes(10)); // Output: true
242 // -----
243
244 // 13. indexOf()
245 console.log(arr8.indexOf(10)); // Output: 1
246 // -----
247
248 // 14. join()
249 console.log(arr8.join('-')); // Output: "5-10-15"
250 // -----
251
252 // 15. map()
253 const mapped = arr8.map(x => x * 2);
254 console.log(mapped); // Output: [10, 20, 30]
255 // -----
```

```
257 // 16. pop()
258 const arr10 = [1, 2, 3];
259 console.log(arr10.pop()); // Output: 3
260 console.log(arr10);       // Output: [1, 2]
261 // -----
262
263 // 17. push()
264 arr10.push(4);
265 console.log(arr10); // Output: [1, 2, 4]
266 // -----
267
268 // 18. reduce()
269 const arr11 = [1, 2, 3, 4];
270 const sum = arr11.reduce((acc, val) => acc + val, 0);
271 console.log(sum); // Output: 10
272 // -----
273
274 // 19. reverse()
275 arr11.reverse();
276 console.log(arr11); // Output: [4, 3, 2, 1]
277 // -----
278
279 // 20. shift()
280 const arr12 = [10, 20, 30];
281 console.log(arr12.shift()); // Output: 10
282 console.log(arr12);         // Output: [20, 30]
283 // -----
284
285 // 21. slice()
286 const arr13 = [1, 2, 3, 4, 5];
287 console.log(arr13.slice(1, 4)); // Output: [2, 3, 4]
288 // -----
289
290 // 22. some()
291 console.log(arr13.some(x => x > 4)); // Output: true
292 // -----
293
294 // 23. sort()
295 const arr14 = [3, 1, 4, 2];
296 arr14.sort();
297 console.log(arr14); // Output: [1, 2, 3, 4]
298 // -----
```

```
299
300 // 24. splice()
301 const arr15 = [1, 2, 3, 4];
302 arr15.splice(1, 2, 9, 8);
303 console.log(arr15); // Output: [1, 9, 8, 4]
304 // -----
305
306 // 25. toString()
307 console.log(arr15.toString()); // Output: "1,9,8,4"
308 // -----
309
310 // 26. unshift()
311 arr15.unshift(0);
312 console.log(arr15); // Output: [0, 1, 9, 8, 4]
313 // -----
314
315 // 27. with() - indexdagi qiymatni almashtirib, yangi massiv qaytaradi
316 const arr16 = [10, 20, 30];
317 const newArr1 = arr16.with(1, 99);
318 console.log(newArr1); // Output: [10, 99, 30]
319
```