

# Ćwiczenia do tablic i obiektów

1. Stwórz funkcję `mergeArrays(array1, array2)`, która zwróci nową tablicę, będącą połączeniem dwóch przekazanych tablic. Np.:

`mergeArrays([2, 5], [3, 7])` powinno zwrócić `[2, 5, 3, 7]`.

Użyj składni *spread*.

2. Stwórz funkcję `exceptFirst()`, która zwróci tablicę ze wszystkimi przekazanymi do niej argumentami oprócz pierwszego. Np.:

`exceptFirst(2, "delfin", null)` powinno zwrócić `["delfin", null]`.

Użyj składni *rest parameter*.

3. Stwórz funkcję `last2Parameters()`, która zwróci tablicę z dwoma ostatnimi argumentami przekazanymi do funkcji. Np.:

`last2Parameters(1, 3, undefined, true)` powinno zwrócić `[undefined, true]`.

Użyj `Array.slice`.

4. Stwórz funkcję `indexOf(element, array)`, która zwróci indeks podanego elementu w podanej tablicy. Np.:

`indexOf("kawa", ["rum", "kawa", "sangría"])` powinno zwrócić `1`.

Użyj `Array.indexOf`.

5. Stwórz funkcję `findNonEmptyTask(tasks)`, która zwróci pierwsze zadanie, które ma treść. Np.:

```
findNonEmptyTask([
  {content: ""},
  {content: "kupić balony w kształcie psów"},
])
```

powinno zwrócić `{content: "kupić balony w kształcie psów"}`.

Użyj `Array.find`.

6. Stwórz funkcję `oddIndex(numbers)`, która zwróci indeks pierwszej nieparzystej liczby z podanej tablicy. Np.:

`oddIndex([2, 4, 7, 8])` powinno zwrócić `2`.

Użyj `Array.findIndex` .

7. Stwórz funkcję `hasStrawberry(fruits)` , która sprawdzi, czy w podanych owocach jest truskawka. Np.:

`hasStrawberry(["banan", "marakuja"])` powinno zwrócić `false` .

Użyj `Array.includes` .

8. Stwórz funkcję `someAdult(people)` , która sprawdzi, czy wśród podanych osób jest ktoś dorosły. Np.:

```
someAdult([
  {name: "Melodia", age: 15},
  {name: "Kosmo", age: 19},
])
```

powinno zwrócić `true` .

Użyj `Array.some` .

9. Stwórz funkcję `onlyString()` , która sprawdzi, czy wszystkie podane argumenty są tekstowe. Np.:

`onlyString("", "żelazko")` powinno zwrócić `true` .

Użyj `Array.every` .

10. Stwórz funkcję `filterPremium(carBrands)` , która z podanej tablicy marek samochodów zwróci tylko marki premium. Np.:

`filterPremium(["Peugeot", "BMW", "Audi"])` powinno zwrócić `["BMW", "Audi"]` .

Założmy, że marki premium to *BMW*, *Audi* i *Mercedes*.

Użyj `Array.filter` oraz `Array.includes` .

11. Stwórz funkcję `getColors()` , która zwróci tablicę kolorów samochodów podanych w argumentach. Np.:

```
getColors(
  {car: "BMW X5", color: "black"},
  {car: "Peugeot 3008", color: "white"},
)
```

powinno zwrócić `["black", "white"]` .

Użyj `Array.map` .

12. Stwórz funkcję `sortPeople(people)` , która posortuje osoby od najmłodszych do najstarszych.  
Przykładowe użycie:

```
sortPeople([
  {name: "Krzychu", age: 30},
  {name: "Zdzichu", age: 20},
])
```

Użyj `Array.sort` .