1. Napisz funkcję, która przyjmuje dwa parametry: tablicę oraz wartość logiczną (true/false). Funkcja powinna zwracać sumę parzystych bądź nieparzystych wartości z tablicy (parzystych, jeśli drugi parametr miał wartość true, zaś nieparzystych w przeciwnym przypadku).
   * Jeśli pierwszym parametrem nie jest tablica, zwróć undefined. Możesz to sprawdzić przy pomocy Array.isArray()
   * Nie należy liczyć sumy, jeśli drugi parametr nie ma wartości ani true, ani false (używaj porównania === lub sprawdź typ wartości przy pomocy typeof)
   * Napisaną funkcję przetestuj na trzy sposoby: na tablicy wartości wpisanych ręcznie (w kodzie), na tablicy wypełnionej losowymi wartościami oraz na tablicy podanej przez użytkownika (jako wartości oddzielane przecinkami).
2. Napisz funkcje, która przyjmuje jako parametry dwie liczby całkowite a i b, a następnie tworzy i zwraca jako wynik tablicę wypełnioną kolejnymi liczbami od a do b.
   * Na przykład, jeśli funkcja zostanie wywołana z argumentami 3 i 8, powinna zwrócić tablicę [3, 4, 5, 6, 7, 8].
   * Funkcja powinna również obsługiwać sytuację, gdy wartość a jest większa niż wartość b, zwracając wtedy pustą tablicę.
3. Przygotuj dokument HTML składający się z pewnej liczby paragrafów (elementów <p>). Następnie (już w kodzie JavaScript):
   * Powiększ rozmiar co drugiego paragrafu (użyj ustawienia stylu fontSize)
   * Tekst każdego paragrafu pokoloruj na losowy kolor (kolory możesz losować ze zdefiniowanej uprzednio puli, np. ‘orange’, ‘green’, ‘blue’, ‘yellow’).
   * Do każdego paragrafu dodaj atrybut title (podpowiedź pojawiająca się po najechaniu myszą) zawierający informację który to paragraf spośród ilu i jaka jest jego długość (w znakach). Np. „Nr 2/10, długość: 54”.
4. Napisz program, który na podstawie tablicy napisów (ścieżek do obrazków – mogą to być obrazki znajdujące się w jakimś podfolderze projektu) utworzy w dokumencie galerię obrazków (elementów <img>).
   * Pod każdym obrazkiem powinna znajdować się jego nazwa (bez ścieżki i rozszerzenia – możesz ją wydobyć wycinając tekst między ostatnim znakiem '.' a ostatnim znakiem '/').
5. Interfejs aplikacji powinien składać się z pola tekstowego oraz przycisków „dodaj”, „usuń” i „resetuj”. Użytkownik wprowadza w polu tekstowym imię, następnie naciskając „dodaj” może je dodać do listy imion (o ile już się na niej nie znajduje) lub naciskając „usuń” – usunąć (o ile jest na liście). Naciśnięcie reset czyści listę.
   * Imiona przechowuj w tablicy (w zmiennej globalnej), a na tej podstawie aktualizuj widok (dokument HTML). Zamiast dodawać lub usuwać elementy HTML po każdej aktualizacji tablicy, prostsze może być odtworzenie widoku od zera.