Финален тест – ниво

# JavaScript

**Вариант А-2019**

*Време за работа – 3 часа.*

*Необходими са 50% от точките за успешно преминаване на теста и 80% за отличие.*

1. Напишете функция, която връща масив с уникални елементи - числа. Функцията получава като параметър - стринг. При подаване на празен стринг , функцията отпечатва в конзолата - The input must not be an empty string. В случай, че стрингът не съдържа числа, функцията отпечатва - ‘The input values must be convertible to numbers’. Функцията ви трябва да работи коректно за посочените по-долу входни данни -

**input**                  **output**

‘’                    The input must not be an empty string

‘1 2 2 6 8 6 7 8’            [1, 2, 6, 8, 7]

‘0 0 0 0 0 ’                [0]

‘mk pp we vfv’            ‘The input values must be convertible to

a number’

‘1 br 2 1 werr 3 1’        [1, 2, 3]

‘2 “” add “” 1 1 asd 2 “” 3’    [2 1 3]

*Оценяване – за всеки успешен тест – 3 т.*

*Общо - 18 т.*

1. Разполагате с index.html, в който има един div с id = ‘wrapper’. Като използвате JS функциите за манипулация на DOM и принципите за създаване и добавяне на нови елементи, добавете в този div меню съставено от 4 li, във всяко от тях има по един a - елемент, с href и текст, указващ накъде води съответния линк.

*Оценяване – 9 т.*

1. **Задача - jQuery**

**Вариант A**

От готовия файл - task\_03\_a.html, като използвате **jQuery** и външен .js файл и без да променяте task\_03\_a.html или да променяте/използвате css, направете валидация на форма, съдържаща полета за въвеждане на

1. име - не трябва да бъде празен стринг.
2. парола - не трябва да бъде празен стринг или да съдържа по-малко от 6 знака.
3. При изпращане на формата с невалидни данни, до съответното поле да се появява съобщение, предупреждаващо за грешката/грешките, допуснати от потребителя.
4. Грешките са оцветени в червено.
5. При успешно попълване на формата, се появява съобщение - “Success!”.

*Оценяване*

* a, b и c подусловия – по 3 точки;
* d - 1 т.
* e - 2 т.

*Общо 12 т.*

**Вариант B**

Като използвате **jQuery** и външен .js файл, без да променяте task\_03\_b.html или да променяте/използвате css, от готовия файл - task\_03\_b.html, който отпечатва една до друга колекция от снимки, при преминаване върху която и да е от снимките с курсора на мишката

1. съответната снимка се отпечатва под редицата снимки с ширина 600 px.
2. над снимката да се отпечатва заглавието й.
3. заглавието е с размер на буквите, различен от стандартния за браузърите.

*Оценяване*

* a - 7 т
* b - 4 т.
* c - 1 т.

*Общо 12 т.*

**Вариант C**

От готовия файл - task\_03\_a.html, като използвате **jQuery** и външен .js файл и без да променяте task\_03\_a.html или да променяте/използвате css, направете валидация на форма, съдържаща полета за въвеждане на

1. данните от формата се изпращат на сървъра
2. данните се валидират на сървъра

* име - не трябва да бъде празен стринг.
* парола - не трябва да бъде празен стринг или да съдържа по-малко от 6 знака.

1. резултатът от валидацията се изпраща в подходящ формат в браузъра.
2. при изпращане на невалидни данни, до всяко поле от формата с невалидни данни, се появява съобщение за грешка /съобщение за грешка се вижда само до полето с невалидни данни/
3. Грешките са оцветени в червено.
4. При успешно попълване на формата, над нея се появява съобщение - “Success!”.
5. Всяко изпращане на данните до сървъра изчиства предишни съобщения.

**Забележка:** Валидацията на формата се извършва на сървъра. Резултата от валидацията се връща в браузъра без презареждане на страницата - използвайте Ajax!

*Оценяване*

* a - 1,5
* b - -
* c - 3
* d - 7,5
* е - 1,5
* f - 3
* g - 1,5

*Общо 18 т.*