МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Практикум №5

з курсу «Основи Front-end технологій»

на тему: «JavaScript. «Події. Регулярні вирази»»

|  |  |
| --- | --- |
| Викладач:  Жереб К.А. | Виконав:  Хільчук А.В.  студент 3 курсу  групи ІП-14 ФІОТ |

Київ-2023

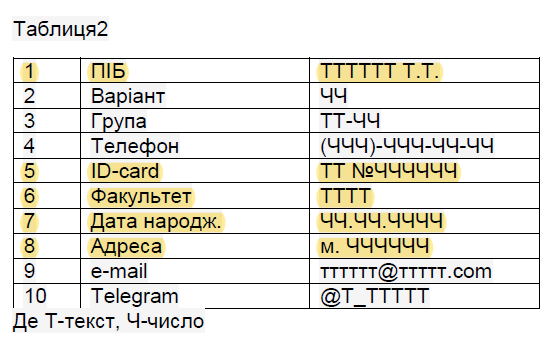
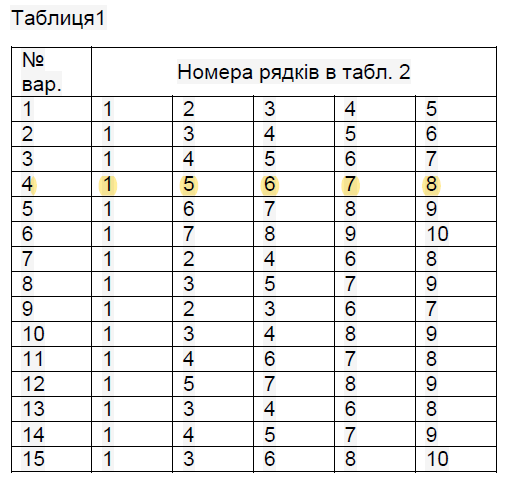
Практична робота №5

**Тема:** JavaScript. «Події. Регулярні вирази»

**Завдання:**

**Завдання1.**

Вибрати рядок у таблиці 1, номер якого збігається з Вашим варіантом. Для формування форми взяти рядки, які відповідають Вашому варіанту таблиці2. З отриманої інформації сформувати форму. За допомогою регулярних виразів JS виконати перевірку валідності введеної інформації. Якщо все правильно – вивести введену інформацію в окремому вікні, якщо є помилки – виділити рядки, які містять помилку.



**Завдання2.**

Створити таблицю розміром (6х6). Клітинки таблиці заповнюються послідовно номерами від 1 до 36 по рядках. При наведенні на клітинку, що відповідає номеру варіантавиконується зміна кольору на випадковий, при Click на ній - зміна кольору на обраний з палітри, а при dblClick - зміна кольору клітинок побічної діагоналі таблиці.

**Виконання:**

*Опис використаних технологій:*Серед нових використаних засобів рзробки в html документі було представлено елементи form та input.

<form> представляє розділ документа, що містить інтерактивні елементи керування для надсилання інформації. Форма може містити в собі теги: <input>, <label>, <select>, <textarea>, тощо. Даний елемент також може виконувати окремі дії після настискання елементу вводу (button, input) з типом submit в його області видимості: це може бути як HTTP запит, так і виконання деякого скрипту на сервері. Документація елементу знаходиться за посиланням: <https://developer.mozilla.org/en-US/docs/Learn/Forms>

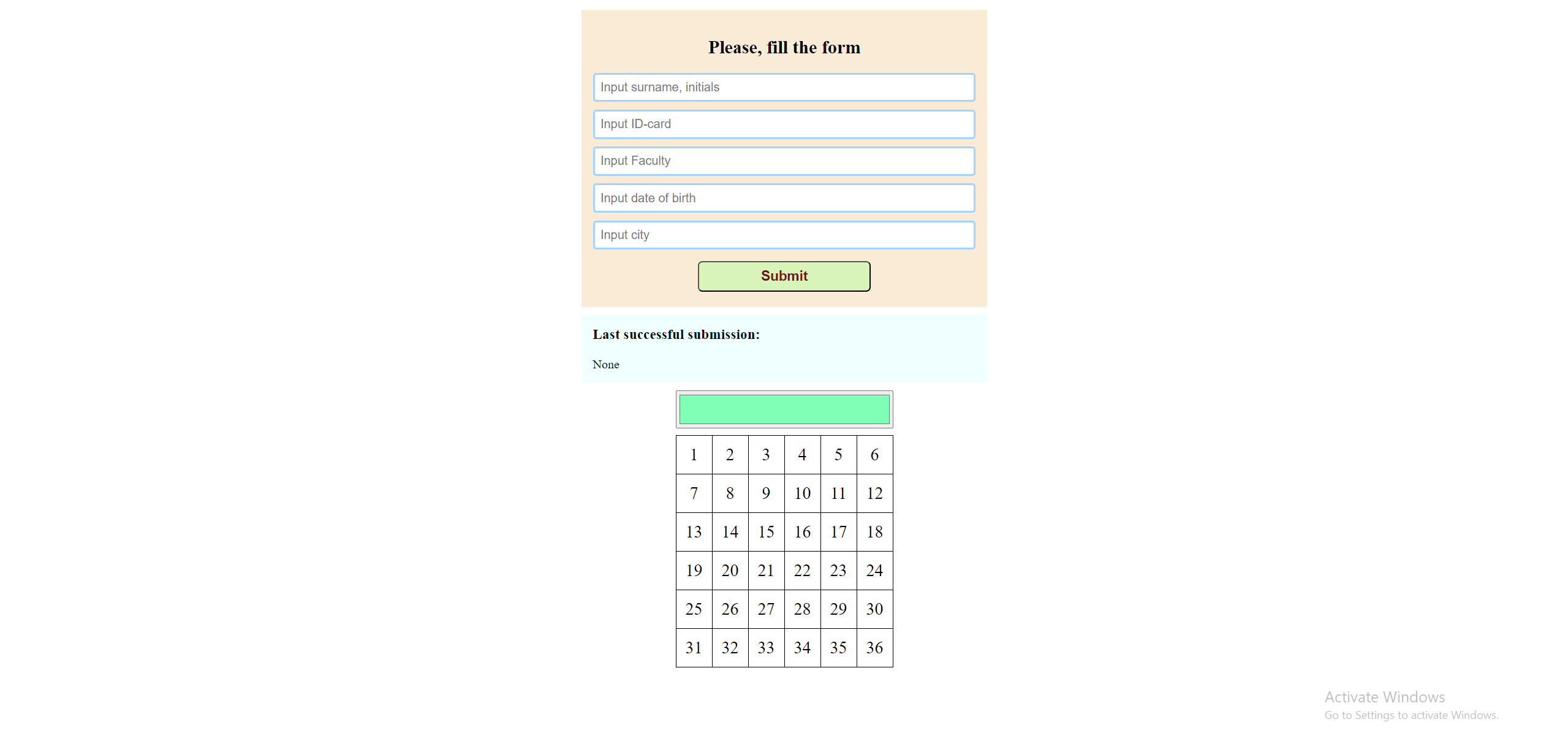
Іншим новим елементом є <input>. Даний елемент використовується для створення інтерактивних засобів керування для веб-форм, щоб приймати дані від користувача. Він має широкий вибір типів вхідних даних: текст, дати, колір, числа, паролі, файли, тощо. З програматичної точки зору, в DOM інтерфейсі такі елементи мають значення value – власне введені значення. У даній роботі використовуватимусться типи саме text та color. Більш детально можна ознаймитися за посиланням: <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/input>

Для валідації вхідних даних форми застосовувалися регулярні вирази. Регулярні вирази — це шаблони, які використовуються для зіставлення комбінацій символів у рядках. У рамках даної ЛР застосвувався клас RegExp та його метод test(). Регулярні вирази задаються стрічками за допомогою стандртної нотації. Конкретно в даній роботі було застосовано такі специфікації: “[A-Z]” та “[a-z]” для великих та малих символів відповідно, “+” та “{n}” для встановлення кількості очікуваних символів, “[0-9]” та “d” для задання чисел, буквальні символи (наприклад, крапки, пробіли, тощо), а також ^ та $ для вказання початку та кінця виразу відповідно. Детальніше можна звернутися за документацією: <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Regular_expressions>

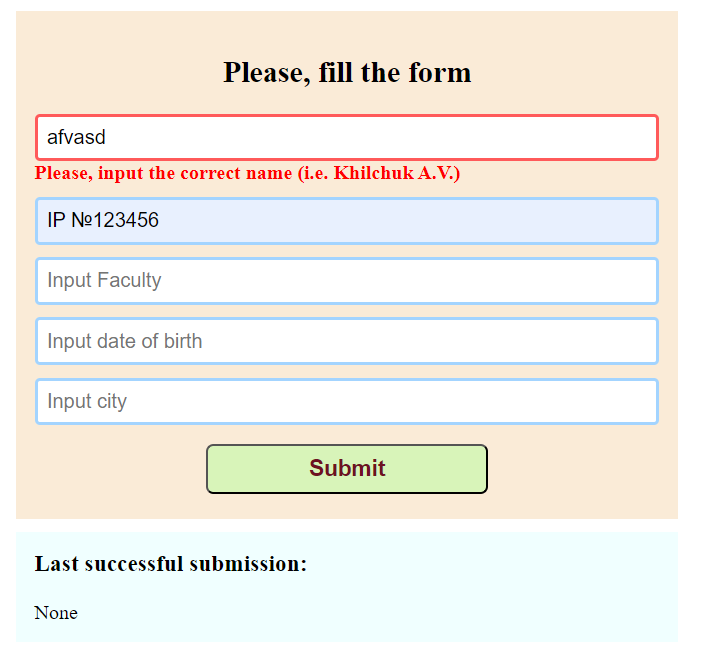
Також було застосовано події. Події запускаються, щоб повідомити код про зміни в стані документу або певні процеси, які можуть вплинути на виконання коду. Вони можуть виникати через такі дії користувача, як натискання на елемент, наведення, відведення, тощо. Подій є надзивчайно широкий спектр, з якими можна ознайомитися за документацією: <https://developer.mozilla.org/en-US/docs/Web/Events>, однак у рамках даної лр буде використано такі: focus, focusout, click, dblclick, mouseOver.

Події можна оброблювати, якщо програматично додавати слухачі події до DOM елементів за допомогою addEventListener() або addEventListeners().

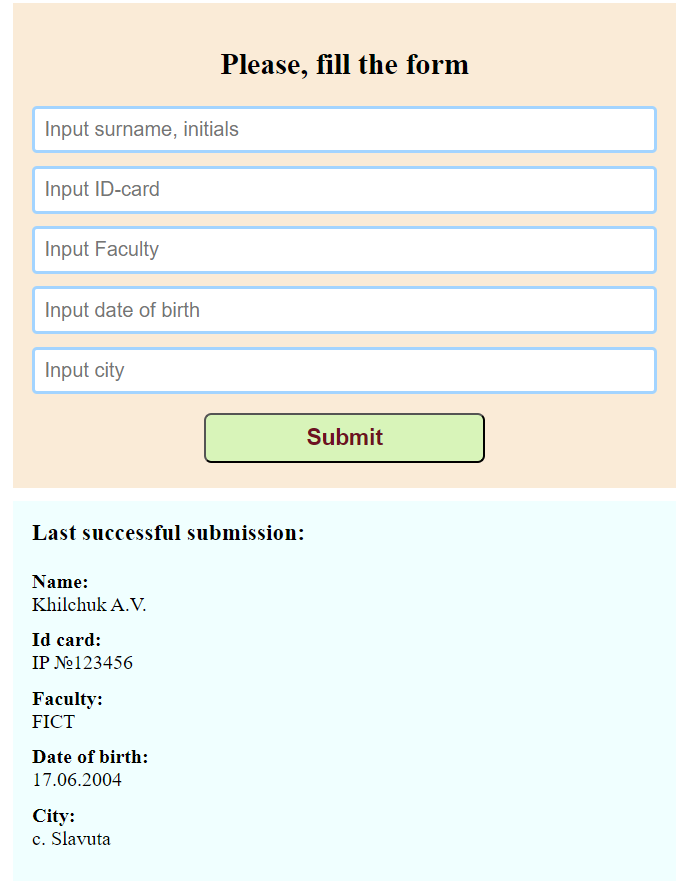
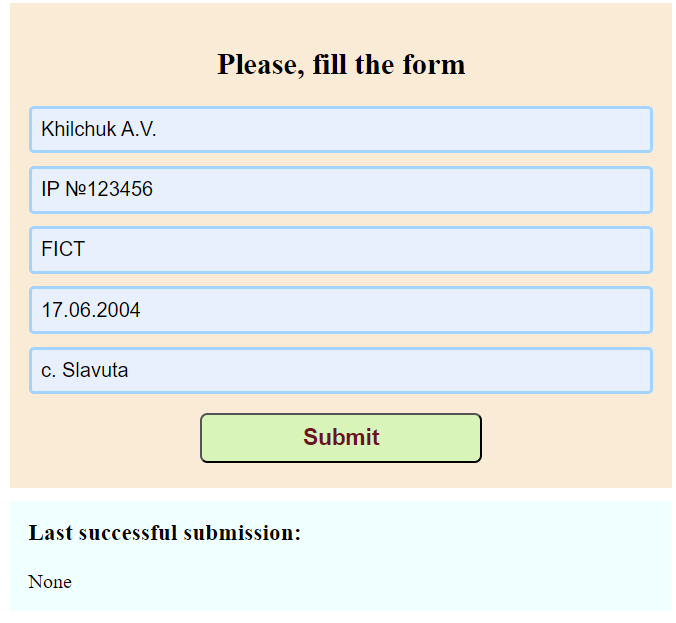
*Результуюча сторінка:*Результатом виконання лабораторної роботи є дана веб-сторінка:



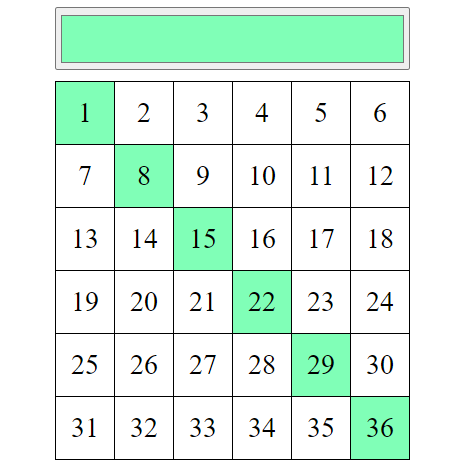
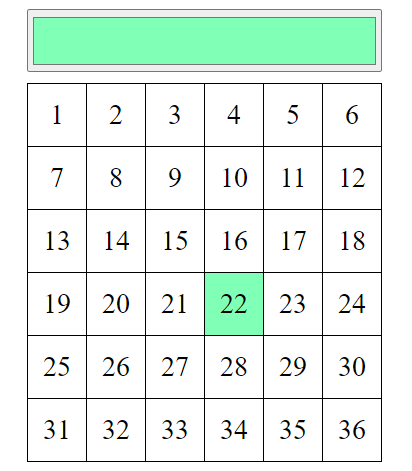
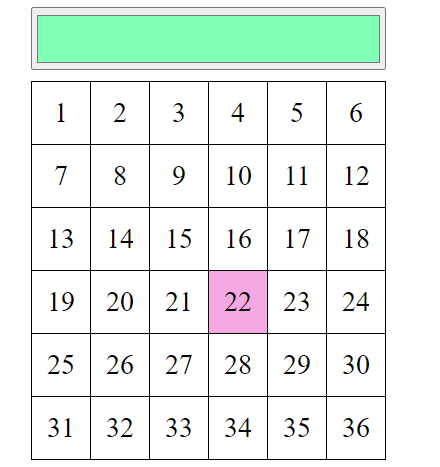
Приклад обробки помилки валідації:



Приклад успішно заповненої форми:



Приклад взаємодії з таблицею(випадковий колір, click, double click):



Сторінку було розгорнуто на Github Pages за даним посиланням: <https://khilchukartemip-14.github.io/web-front-end-basics/Lab5/Index.html>

Із кодом можна ознайомитися в даному репозиторії:

[https://github.com/KhilchukArtemIP-14/web-front-end-basics/tree/main/Lab5/Index.html](https://github.com/KhilchukArtemIP-14/web-front-end-basics/tree/main/Lab5/Index.html%20)

**Висновок**

Отож, у ході виконання лабораторної роботи було зверстано сторінку, що містить у собі форму, котра по заповненню валідує введені дані виразів та в разі успіху зберігає результат в окремому вікні, а також кольорову палітру та таблицю, що при наведенні змінює колір 22 клітинки на витпадковий, при натисканні - на колір зафіксований в палітрі, а при подвійному натисканні зафарбовує фіксованим кольором побічну діагональ. Таку функціональність було досягнуто за допомогою використання регулярних виразів, подій та їх обробки, а також елементів <form> та <input>. Урешті-решт, сторінку було розгорнуто на хостингу Github Pages. Набуто практичних навичок застосування JavaScript для створення та валідації форм, а також обробки різних подій.