**Лабораторна робота 10. Зберігання даних. Протокол HTTP. AJAX. Кібербезпека**

1. **Введення повідомлення**

Попросіть користувача ввести повідомлення (prompt) і збережіть його в *localStorage*. При завантаженні сторінки підставляйте збережене значення з *localStorage* у якості значення за замовчанням при запиті введення (другий аргумент функції prompt).

1. **Поле введення**

Зробіть те ж, що у попередній задачі (введення повідомлення), тільки замість стандартного *prompt* використовуйте текстове поле введення та кнопку.

1. **JSON**

Створіть звичайний об’єкт із властивостями “name”, “secondName” та “email” і деякими значеннями за замовчанням, і запишіть його у будь-яку змінну. Спробуйте вивести цей об’єкт за допомогою *alert*. Зміст об’єкта не виводиться, так як інтерпретатор не перетворює цей зміст у рядок. Щоб його вивести, використовуйте *JSON.stringify(object)* для перетворення об’єкту на рядок. Якщо все правильно, ви повинні побачити всі властивості об’єкта та їх значення в повідомленні.

1. **Форма логіна**

Створіть форму з полями введення для логіна та пароля і кнопкою для відправки форми. Дані форми повинні відправлятися методом POST за адресою “/users/login”.

*Підказка:* використовуйте атрибути форми для ставлення методу та адреси відправки. Задайте відповідні імена полям форми. Перевірте правильність відправки запиту, використовуючи вкладку *Network* інструментів розробника (не забудьте поставити галочку “Preservelog”, щоб запит відображався і після перезавантаження сторінки через відправку форми).

1. **Привітання**

При завантаженні сторінки виведіть на екран *alert* із текстом «Доброго ранку!», «Добрий день!» або «Добрий вечір!» в залежності від поточного часу. Користувач повинен бачити таке повідомлення не частіше, ніж раз у день.

*Підказка*: для вибору повідомлення використовуйте умовні оператори, порівнюючи поточне значення годин (метод *getHours()* об’єкту дати) із значеннями 12 і 18 (до 12 – ранок, після 18 – вечір, між ними - день). Щоб запам’ятати, коли користувач бачив повідомлення, збережіть у *localStorage* поточне значення дати. При завантаженні сторінки зчитуйте це значення, і якщо ще не настав наступний від збереженої дати день, нічого не виводьте, а інакше (у тому числі якщо жодна дата не була збережена) – виведіть повідомлення та збережіть нову дату на місце попередньої в *localStorage.*

1. **ПІБ**

Створіть конструктор *Person*, який приймає 3 параметри: ім’я, прізвище та по батькові, і створює об’єкт із властивостями *firstName, secondName, lastName*. Додайте створюваному об’єкту метод *fullName()*, який повертає повне ім’я у вигляді рядку, складеного з цих трьох значень. Викликаючи даний конструктор, створіть декілька таких об’єктів для прикладу.

1. **Калькулятор**

Створіть конструктор *Calculator*, який створює об’єкт із наступними методами:

* **read()** – виводить два *prompt’а*, у яких зчитує два числових значення та записує їх у об’єкт
* **sum()** – виводить *alert* із сумою введених значень
* **sub()** – виводить *alert* із різницею введених значень
* **mul()** – виводить *alert* із добутком введених значень
* **div()** – виводить *alert* із часткою введених значень

Кожен метод повинен вертати посилання на даний об’єкт (тобто *this*).

Далі створіть один калькулятор, викличте його метод *read(),* а після ланцюжком викличте останні методи. Тобто, наприклад, повинен спрацювати такий код:

**var calc = new.Calculator();**

**calc.read().sum().sub().mul().read().div();**