**ОТЧЕТ**

**о выполнении лабораторной работы №8**

**Дисциплина «МДК 03.01 Моделирование и анализ программного обеспечения»**

**Тема: «Проверки на стороне клиента»**

**Специальность: 09.02.07 Информационные системы и программирование**

**Квалификация: Специалист по информационным системам**

Выполнил: Стадников М.А.

Студент группы: ИС50-2-22

Проверил:

Преподаватель: Кретова Е.М.

Дата: «\_\_\_» \_\_\_\_\_\_\_\_\_ 2024 г.

Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Теоретические сведения**

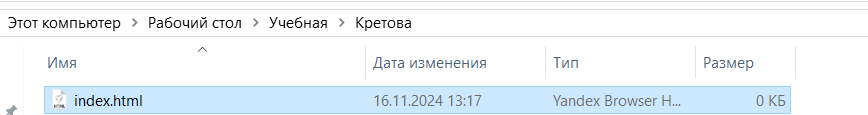
**Проверка на стороне клиента (или валидация на стороне клиента)** — это процесс верификации данных, введенных пользователем, до их отправки на сервер. Эта проверка осуществляется с помощью различных технологий и инструментов, таких как JavaScript, HTML5 и CSS.

Основные аспекты проверки на стороне клиента:

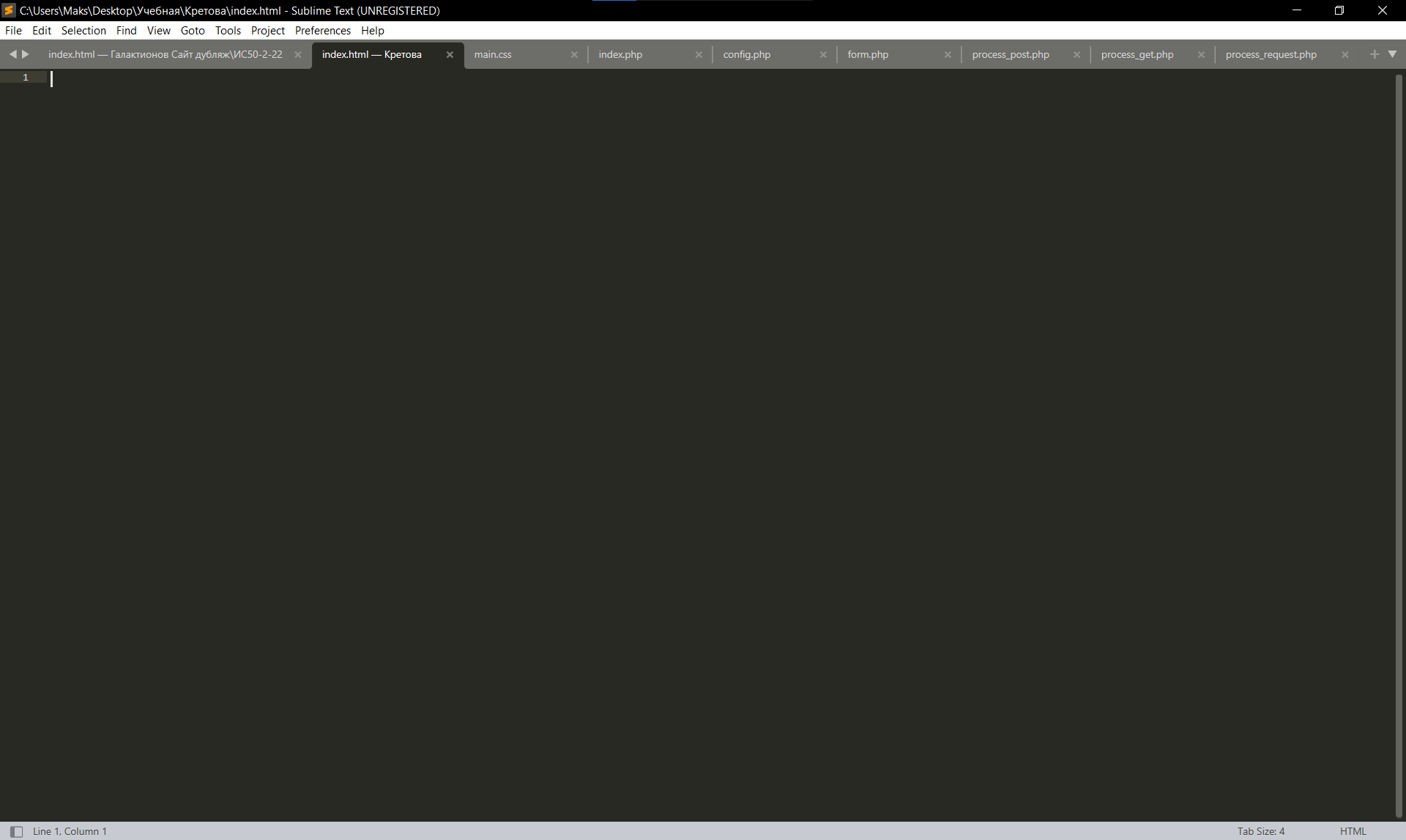
1. Цели проверки:
   * Обеспечение правильности введенных данных.
   * Улучшение пользовательского опыта, уменьшая количество ошибок и предоставляя пользователю мгновенную обратную связь.
   * Снижение нагрузки на сервер, так как невалидные данные могут быть отклонены до отправки.
2. Методы проверки:
   * HTML5 валидация: Современные браузеры поддерживают встроенные возможности валидации форм через атрибуты, такие как required, pattern, min, max и т.д.
   * JavaScript валидация: Использование скриптов для проверки введенных данных. Это может быть, как простая проверка (например, проверка на наличие пустых полей), так и более сложная (например, проверка формата электронной почты или паролей).
3. Типы проверки:
   * Синтаксическая проверка: Проверка на соответствие определенному шаблону (например, формат даты или email).
   * Логическая проверка: Проверка на соответствие логическим условиям (например, дата начала не может быть позже даты окончания).
   * Диапазонные проверки: Проверка числовых значений на соответствие заданным пределам.
4. Преимущества:
   * Улучшение пользовательского опыта за счет быстрой обратной связи.
   * Снижение количества запросов на сервер и экономия ресурсов.
   * Уменьшение вероятности ошибок при вводе данных.
5. Недостатки:
   * Зависимость от скриптов: если пользователь отключил JavaScript, валидация не сработает.
   * Безопасность: полагаться только на проверку на стороне клиента небезопасно, так как данные можно манипулировать. Поэтому всегда необходимо дублировать проверку на стороне сервера.
6. Рекомендации:
   * Использовать валидацию на стороне клиента как дополнение к валидации на стороне сервера, а не как единственный способ защиты.
   * Обеспечить доступность форм и вовлекать всех пользователей, включая людей с ограниченными возможностями.
   * Обновлять и тестировать систему валидации в соответствии с изменениями в требованиях и спецификациях.

**Порядок выполнения работы**

1. Создаем файл index.html

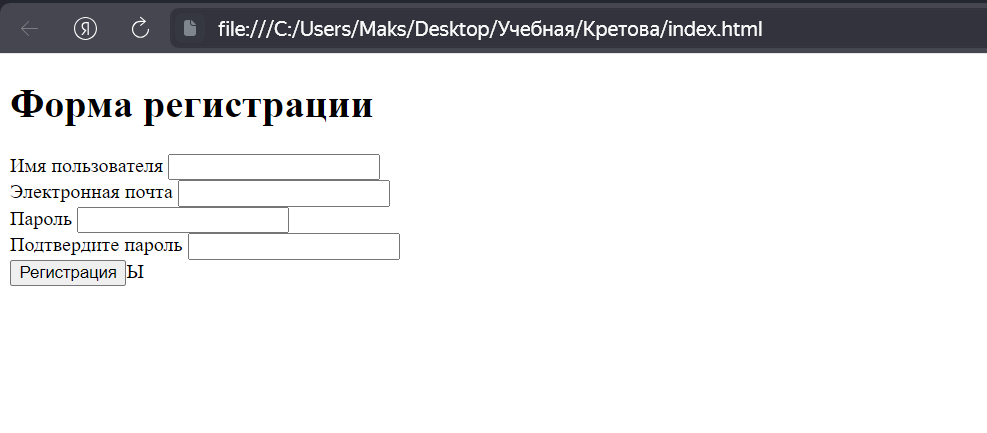


1. Открываем файл в любом редакторе кода



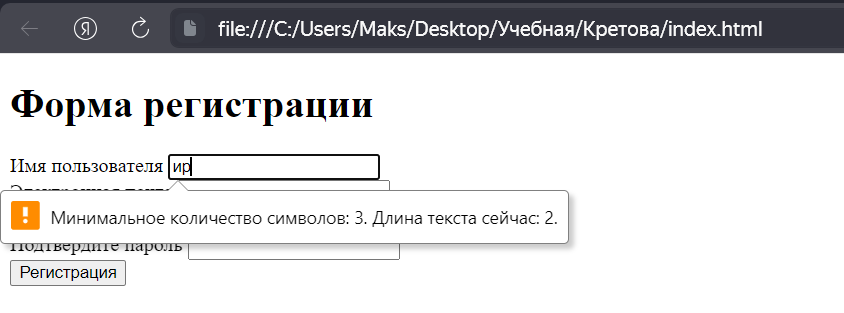
1. Прописываем следующий код в файле index.html



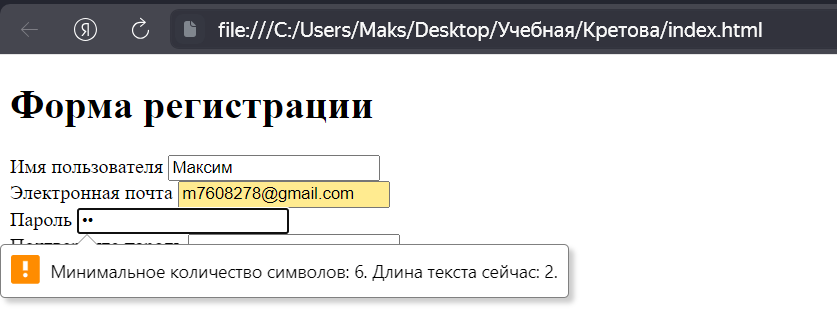


1. Необходимо протестировать следующие проверки валидации

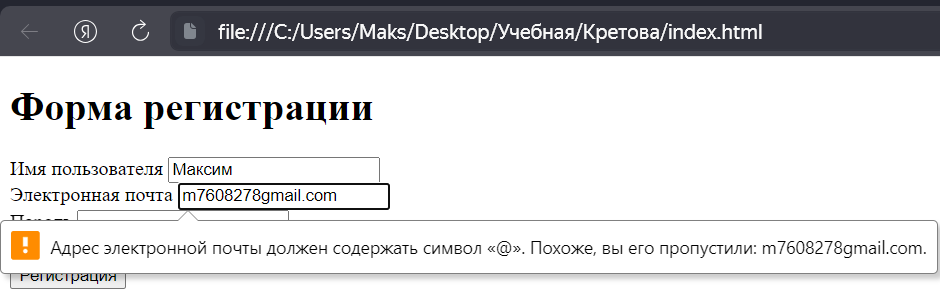
Проверка на допустимую длину имени пользователя



Проверка на допустимую длину длину пароля



Проверка на ввод допустимого адреса электронной почты



Проверка на заполнение обязательных полей ввода

