

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН**

Учреждение образования «Центральноазиатский технико-экономический колледж»

ОТЧЕТ

По учебной практике:

Применение систем управления веб-контентом

Выполнил:

студент гр. P2A

Fedorov Ilya

Проверила:

Жүсіпбекқызы К

Оценка \_\_\_\_\_\_\_\_\_\_\_\_

Алматы 2022

Source code and description: <https://github.com/lyarov22/php-projects>

0.

# Практическая работа №1

# Установка локального веб-сервера

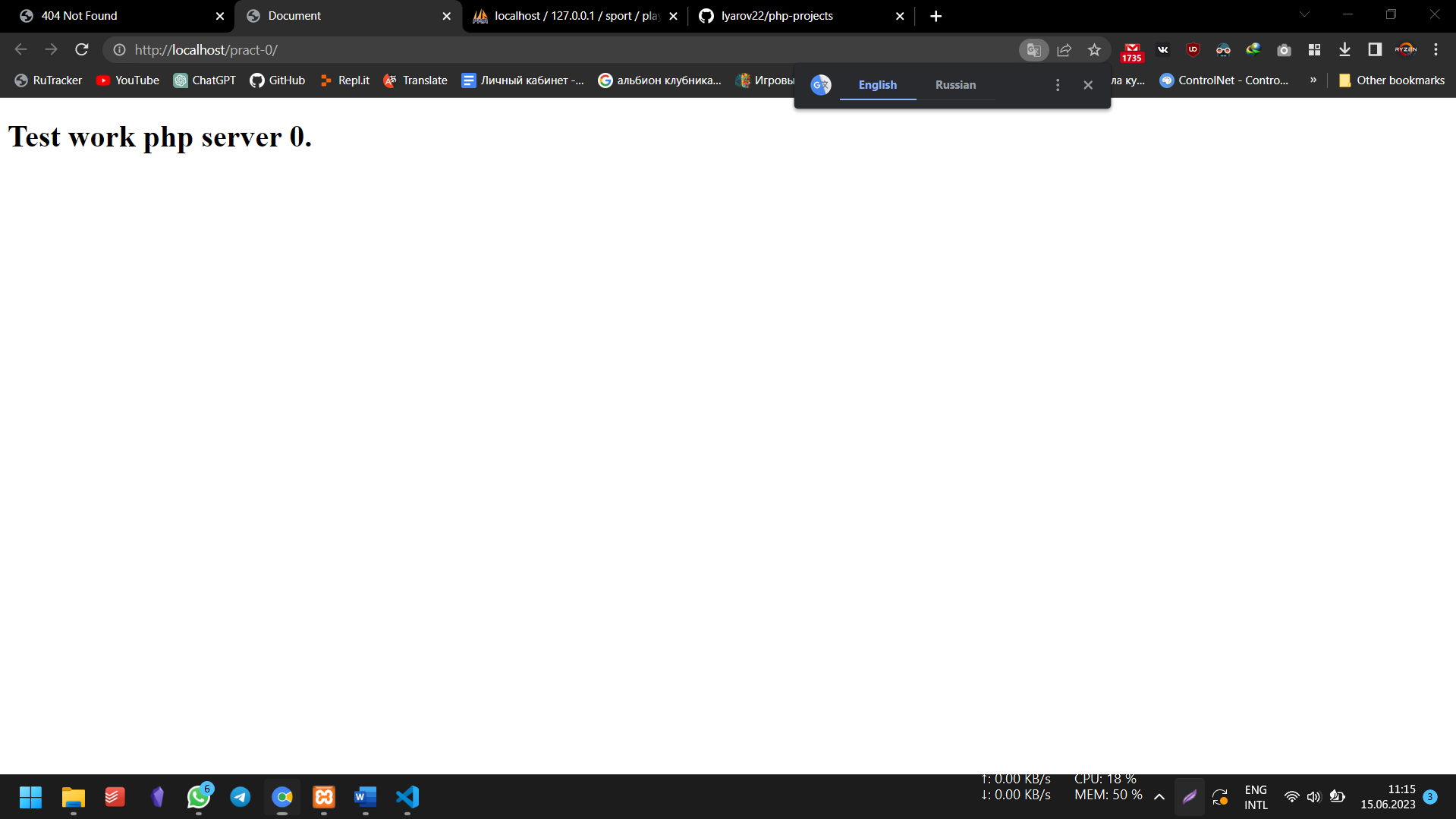
**Цели:**

1. Ознакомиться с работой локального.веб-сервера

**Ход работы:**

1. Идём на сайт <https://www.apachefriends.org/ru/index.html>
2. Качаем XAMPP для Windows:
3. Запускаем скачанный файл от имени администратора:
4. Жмём «Next» до следующего окошка:
5. Оставляем галки на Apache, MySQL, PHP, phpMyAdmin:
6. Жмём «Next», выбираем путь установки. Я оставлю по умолчанию «C:\xampp».
7. Язык оставляем английский.
8. Убираем галку «Learn more about Bitnami for XAMPP».
9. Жмём «Next», ждём, пока установка завершится.
10. Жмём «Finish». Должна запуститься панель управления веб-сервером:
11. Жмём «Start» справа от Apache.
12. Переходим обратно в браузер, в адресную строку пишем «http://localhost».
13. Если открывается такое окошко, то всё установилось нормально:
14. Создаем в C:\xampp\htdocs новую папку «pract».
15. В папке создаём файл index.html.
16. Идём в браузер, пишем в адресную строку «localhost/pract». Если показалось следующее, то всё работает правильно:

Code: <https://github.com/lyarov22/php-projects>/pract0



1.Ознакомление с синтаксисом PHP

**Цели:**

1. **Типы данных PHP**
2. Извлечение символов из строки

**Теория:**

PHP является языком [динамической типизации](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%BD%D0%B0%D0%BC%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D1%82%D0%B8%D0%BF%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F" \t "_blank), тип переменной определяется на основе её значения. Но все же нужно знать какие типы данных существуют в PHP. Ниже перечислены все типы, которые можно использовать в PHP:

* **Boolean**. Это логический тип, который содержит значение true или false.
* **Integer**. Содержит значения целого числа (Например: 4 или 10 или другое целое число).
* **String**. Содержит значение текста произвольной длины (Например: Олег, Киев, Австрия).
* **Float**. Вещественное число (Например: 1.2, 3.14, 8.5498777).
* **Object**. Объект.
* **Array**. Массив.
* **Resource**. Ресурс (Например: файл).
* **NULL**. Значение NULL.

Для предотвращения появления ошибок рекомендуется не смешивать разные типы данных.

Если вы хотите явно изменить тип данных переменной, то для этого нужно слева от имени переменной в круглых скобках указать нужный тип:

**Извлечение символов из строки**

### Типы данных - Задачи по строкам

Задача №1 - Создайте переменную text и присвойте ей значение "abcde".  
Обращаясь к отдельным символам строки нужно вывести на экран символы "a" "c" и "e"

Условия в последующих задачах просты и можно самостоятельно догадаться, что в них делается.

**Самостоятельная работа**

**print\_r("php"[1])**

Укажите нужные индексы, чтобы программа вывела на экран имя Jim:

Начало формы

$name1 = "John"

$name2 = "Tom"

$name3 = "Emily"

print\_r($name1[] . $name3[] . $name2[]) // => JimКонец формы

1. Вам даны три переменные с фамилиями разных людей. Составьте и выведите на экран слово из символов в таком порядке:

1. Третий символ из первой строки
2. Второй символ из второй строки
3. Четвертый символ из третьей строки
4. Второй и третий символ из второй строки

2. Дана строка 'world'. Обращаясь к отдельным символам этой строки выведите на экран символ 'o', символ 'l', символ 'd'.

3. Дана переменная со строкой 'abcde'. Обращаясь к отдельным символам этой строки запишите в новую переменную символы этой строки в обратном порядке, то есть 'edcba'.

4. Создайте переменные, присвоив им логические значения true и false, вещественное число 43.658+e2, но предварительно записав его в обычном виде, и корень квадратный из отрицательного числа, например, sqrt(-2). Выведите значения всех переменных на экран.

5. Создайте три переменные и присвойте им строки: 'Добро ', 'побеждает ', 'зло'. Создайте еще одну переменную и присвойте ей объединенную строку (для конкатенации строковых переменных используйте обычную точку). Выведите ее на экран.

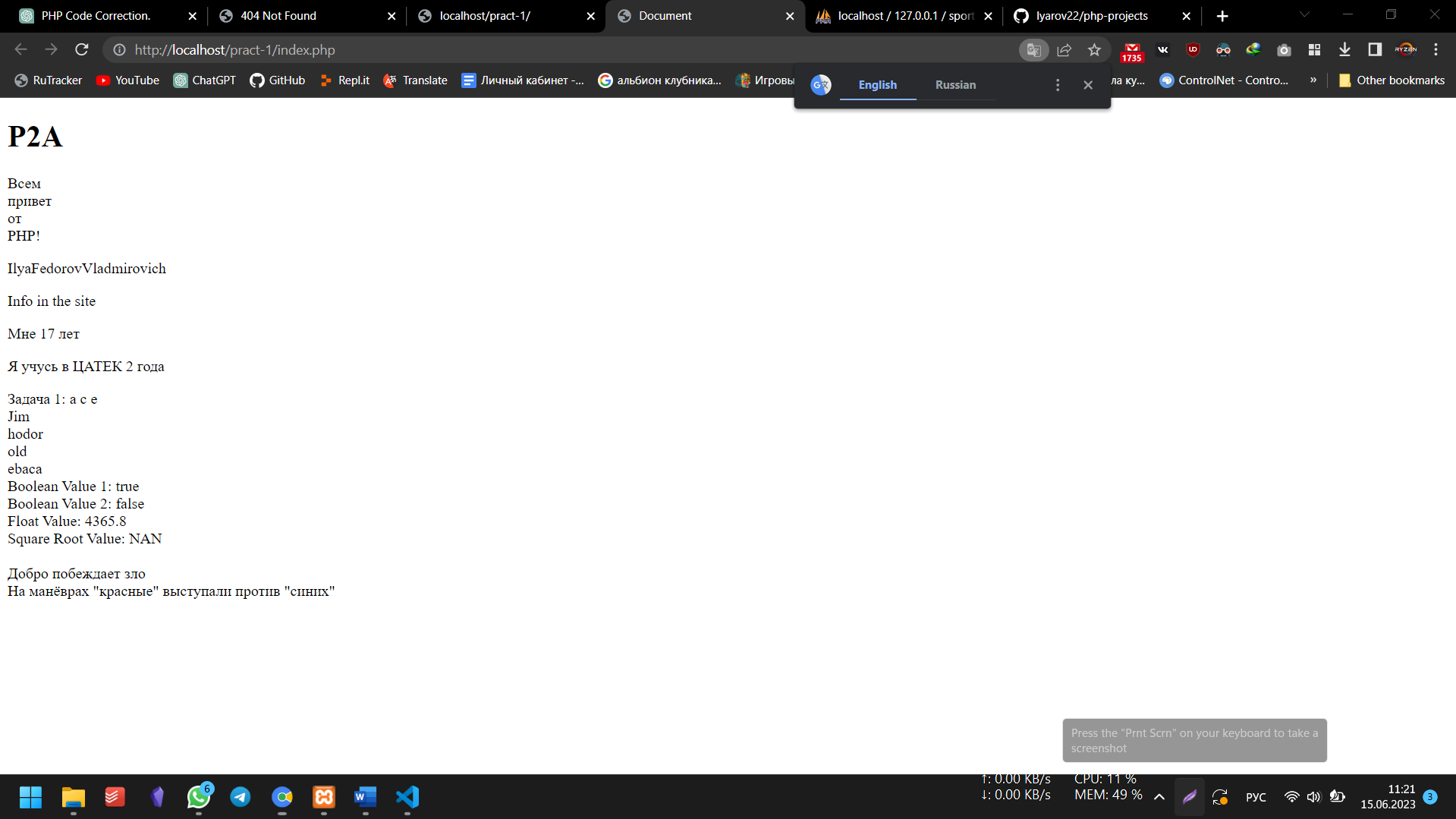
6. Создайте переменную и присвойте ей строку "На манёврах "красные" выступали против "синих"", предварительно экранировав двойные кавычки внутри двойных во избежание ошибки. Выведите строку на экран. Как еще можно поступить для вывода строк, содержащих внутри себя двойные или одинарные кавычки, чтобы не использовать их экранирование?

1. Выведите на экран одна за другой четыре строки: 'Всем ', 'привет ', 'от ', 'PHP!<br><br>', используя для этого только один оператор echo. После данной инструкции напишите два однострочных комментария: 'Для вывода слов на разных строках мы используем элемент "br", т.е разрыв строки' и 'Переменные выводимые оператором echo перечисляются через запятую'. Далее измените код первой части так, чтобы слова после вывода на экран находились на разных строках. В самом конце нашего кода напишите многострочный комментарий: 'После каждой инструкции мы будем ставить точку с запятой даже, если инструкция является последней в блоке php-кода'.
2. Выведите на экран друг за другом три имени: Robert, Stannis, Renly. В результате на экране должно отобразиться:

RobertStannisRenly

1. Создайте переменные с названиями title и content. Присвойте им какие-нибудь значения. Выведите значение переменной title внутри тега h1, а значение переменной content внутри тега div.
2. Выведения фразы «Мне … лет», где количество лет вычисляется на основании текущего года и года рождения.
3. Выведения фразы «Я учусь в ЦАТЭК … года», где количество лет вычисляется на основании текущего года и года поступления.

Code: Code: <https://github.com/lyarov22/php-projects>/pract1



2.Математические операции с числами в PHP

**Цели:**

Атрибуты тегов в PHP

Практика на операции в PHP

**Теория:**

1. С помощью тега <img> выведите на экран какую-нибудь картинку.
2. С помощью тега <input> выведите на экран инпут с каким-нибудь текстом.
3. С помощью тега <textarea> выведите на экран многострочное поле ввода с каким-нибудь текстом.

**Практика на операции в PHP**

**Теория:**

Давайте узнаем сколько секунд в часе. Как известно, в минуте 60 секунд, а в часе - 60 минут. Следовательно, чтобы найти количество секунд в часе, необходимо 60 умножить на 60:

<?php

echo 60 \* 60; // количество секунд в часе

?>

**Самостоятельная работа**

На базовом уровне компьютеры оперируют только числами. Для сложения двух чисел в математике мы пишем, например, 3 + 4. В программировании — то же самое.

**<?php**

***// Снова не забываем точку с запятой в конце строчки***

**print\_r(3 + 4); *// => 7***

Кроме сложения доступны следующие операции:

* \* — умножение
* / — деление
* - — вычитание
* % — [остаток от деления](https://ru.wikipedia.org/wiki/%D0%94%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5_%D1%81_%D0%BE%D1%81%D1%82%D0%B0%D1%82%D0%BA%D0%BE%D0%BC)
* \*\* — возведение в степень

Эти знаки операций называют операторами.

**<?php**

**print\_r(8 + 2);**

Здесь + — это оператор, а числа 8 и 2 — это операнды.

**Композиция операций**

Разберем пример композиции операций:

**<?php**

print\_r(2 **\*** 4 **\*** 5 **\*** 10); *// 2 \* 4 \* 5 \* 10 = 8 \* 5 \* 10 = 40 \* 10 => 400*

Операции можно соединять друг с другом. Это дает возможность вычислять более сложные составные выражения. Такое свойство операций называется **композицией**.

Чтобы представить, как происходят вычисления внутри интерпретатора, разберем пример: 2 \* 4 \* 5 \* 10.

1. Сначала вычисляется 2 \* 4 и получается выражение 8 \* 5 \* 10
2. Затем 8 \* 5. В итоге имеем 40 \* 10
3. В конце происходит последнее умножение, и получается результат 400

### Операции с числами

В данном уроке PHP рассмотрим работу с числами.

**Самостоятельная работа**

1. Создайте переменную $a. Запишите в нее сумму чисел 1, 2 и 3. С помощью команды echo выведите на экран содержимое переменной $a.
2. Создайте переменную $a со значением 10 и переменную $b со значением 2. Выведите на экран их сумму, разность, произведение и частное (результат деления).
3. Создайте переменную $c со значением 10 и переменную $d со значением 5. Просуммируйте их, а результат присвойте переменной $res. Выведите на экран значение переменной $res.
4. Создайте переменную $a со значением 1, переменную $b со значением 2 и переменную $c со значением 3. Выведите на экран их сумму.
5. Создайте переменную $a со значением 10 и переменную $b со значением 5. Отнимите от $a переменную $b и результат присвойте переменной $c. Затем создайте переменную $d, присвойте ей значение 7. Сложите переменные $c и $d, а результат запишите в переменную $res. Выведите на экран значение переменной $res.
6. Напишите скрипт, который:  
   1. Создаёт переменную $price и присваивает ей значение 1500.  
   2. Создаёт переменную с названием $shipping (стоимость доставки) и присваивает ей значение 200.  
   3. Создаёт переменную с названием $name и присваивает ей значение Автомат АК47.  
   4. Выводит 3 строки: название товара, стоимость товара и стоимость товара вместе с доставкой
7. В этом году зарплата Васи составляет 25 000р, а зарплата Пети - 30 000р. В следующем году зарплата Васи вырастет на 20%, а зарплата Пети - на 15%.

Напишите скрипт, который выведет на экран размеры зарплат обоих сотрудников в следующем году.

<?php

$salary\_vasya = 25000;

$salary\_petya = 30000;

?>

1. Компания выпустила 1000 акций. Джон выкупил 100 акций, Столерман - 150, а Иван - 235.

Напишите скрипт, который:  
1. Показывает, сколько процентов акций (от общего количества) имеет каждый из владельцев.  
2. Показывает общее количество проданных и непроданных акций.

1. Напишите программу, которая считает и последовательно выводит на экран значения следующих математических выражений:

* 3 в степени 5
* -8 разделить на -4
* остаток от деления 100 на 3

10. Напишите программу, которая берет исходное количество евро, записанное в переменную eurosCount, переводит евро в доллары и выводит на экран. Затем полученное значение переводит в рубли и выводит на новой строчке.

Пример вывода для 100 евро:

125

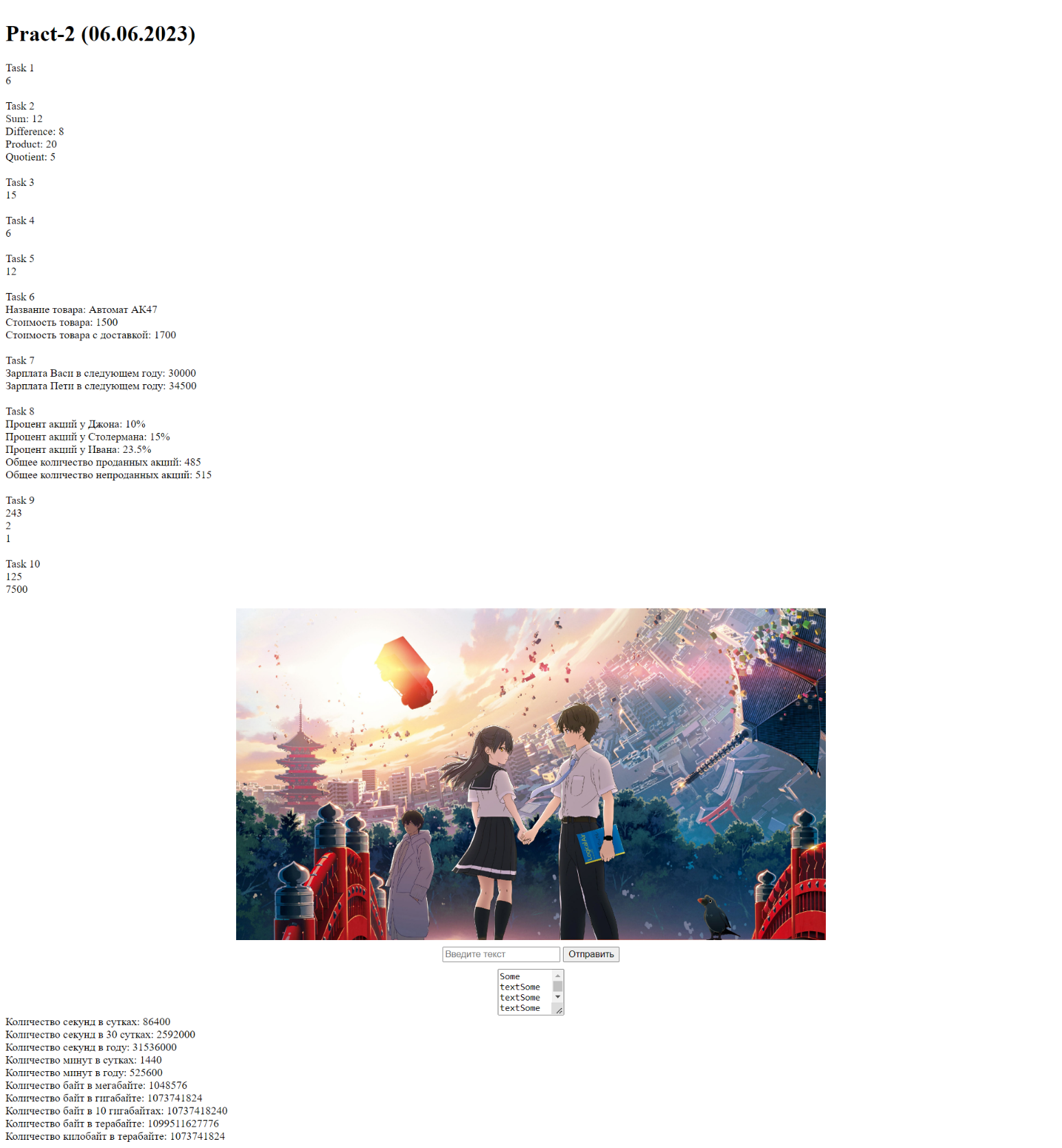
7500

Считаем, что:

* 1 евро = 1.25 долларов
* 1 доллар = 60 рублей

1. Найдите количество секунд в сутках.
2. Найдите количество секунд в 30 сутках.
3. Найдите количество секунд в году.
4. Найдите количество минут в сутках.
5. Найдите количество минут в году.
6. Найдите количество байт в мегабайте.
7. Найдите количество байт в гигабайте.
8. Найдите количество байт в 10 гигабайтах.
9. Найдите количество байт в терабайте.
10. Найдите количество килобайт в терабайте.

Code: <https://github.com/lyarov22/php-projects>/pract2



3.**Отправка форм на сервер**

**Цели:**

Отправка форм на сервер

Работа с полями ввода форм

**Теория:**

Одним из основных способов передачи данных веб-сайту является обработка форм. Формы представляют специальные элементы разметки HTML, которые содержат в себе различные элементы ввода - текстовые поля, кнопки и т.д. И с помощью данных форм мы можем ввести некоторые данные и отправить их на сервер. А сервер уже обрабатывает эти данные.

Создание форм состоит из следующих аспектов:

Создание элемента <form><form> в разметке HTML

Добавление в этот элемент одно или несколько поле ввода

Установка метода передачи данных. Чаще всего используются методы GET или POST

Установка адреса, на который будут отправляться введенные данные

POST-запросы

Итак, создадим новую форму. Для этого определим новый файл form.php, в которое поместим следующее содержимое:

Атрибут action="user.php" элемента form указывает, что данные формы будет обрабатывать скрипт user.php, который будет находиться с файлом form.php в одной папке. А атрибут method="POST" указывает, что в качестве метода передачи данных будет применяться метод POST.

Теперь определим файл user.php, который будет иметь следующее содержание:

Для обработки запросов типа POST в PHP используется встроенная глобальная переменная $\_POST. Она представляет ассоциативный массив данных, переданных с помощью метода POST. Используя ключи, мы можем получить отправленные значения. Ключами в этом массиве являются значения атрибутов name у полей ввода формы.

Например, так как атриут name поля ввода возраста имеет значение age (<input type="number" name="age" />), то в массиве $\_POST значение этого поля будет представлять ключ "age": $\_POST["age"]

И поскольку возможны ситуации, когда поле ввода будет не установлено, то в этом случае желательно перед обработкой данных проверять их наличие с помощью функции isset(). И если переменная установлена, то функция isset() возвратит значение true.

Теперь мы можем обратиться к скрипту form.php и ввести в форму какие-нибудь данные:

И по нажатию кнопки введенные данные методом POST будут отправлены скрипту user.php:

Необязательно отправлять данные формы другому скрипту, можно данные формы обработать в том же файле формы. Для этого изменим файл form.php следующим образом:

Поскольку в данном случае мы отправляем данные этому же скрипту - то есть по тому же адресу, то у элемента форма можно не устанавливать атрибут action.

**Самостоятельная работа**

## 1. Простая форм

Создайте форму с одним полем ввода с запросом любимого города пользователя. Используйте для формы метод "post".

После того, как пользователь щелкнет кнопку отправки, выведите название города, введенное пользователем, сразу же под кнопкой "Отправить".

Подсказка: переменная, содержащая пользовательский ввод, является массивом. Переменная массива - $\_POST['name'], где 'name' - имя вашего поля ввода.

## 2. Интерактивная форма с утверждением If-Else

Возьмите за основу форму с одним полем ввода из предыдущего примера. С помощью условного оператора if-else отобразите, в случае заполнения поля ввода и нажатия кнопки "Отправить", название введенного города. Если поле не заполнено - отобразите форму.

Подсказка: вам понадобится способ узнать, отправлена ​​ли форма. Используйте функцию isset() для определения того, установлена ​​ли переменная и не является ли она нулевым значением.

## 3. Интерактивная форма с конструкцией If-Elseif-Else

В этом упражнении PHP вы будете использовать тот же формат, что и в предыдущем упражнении, запрашивая ввод в первой части скрипта и отвечая во второй с помощью оператора if-else PHP. В первом разделе создайте выпадающий список с днями недели и попросите пользователя выбрать день недели.

Для второй части скрипта, в зависимости от выбранного дня выведите соответствующую строку:  
"Понедельник - 1-й день недели"  
"Вторник это день второй"  
А за ним идет – среда"  
"За средой идет – четверг"  
"Пятница ударница"  
"День шестой – суббота"  
"Воскресенье - полный выходной"

Используя конструкцию else-elseif-else, настройте каждую строку для вывода в ответ на день выбора пользователем.

4. Создайте форму, состав и внешний вид полей которой показан на скриншоте. Скрипт, который будет обрабатывать данные формы на сервере, должен выводить либо сообщения с данными формы либо просить заполнить все поля формы (см. скриншот задачи). Для проверки заполнения полей имени и фамилии используйте условный оператор if и функцию mb\_strlen(). Не забывайте, что после скачивания файла содержимое папки нужно разместить в корневой папке test нашего учебного сайта на локальном сервере (иначе скрипт работать не будет).

5. Создайте в корневой папке test страницу с такой формой, добавив в нее кнопку «Сбросить». Что касается php-скрипта, то его нужно усложнить. Скрипт должен делать проверку заполнения полей имени и фамилии и в случае ошибки прерывать дальнейшее выполнение скрипта и выводить сообщение. Также скрипт должен проверять, был ли сделан выбор предпочтения и в зависимости от этого делать соответствующий вывод (см. скриншот).

1. Создайте форму, состав и внешний вид полей которой показан на скриншоте. Скрипт, который будет обрабатывать данные формы на сервере, должен проверять заполнение полей имени и фамилии и в случае ошибки выводить предупреждение и прерывать скрипт. Что касается аватарки, то скрипт должен либо вывести аватарку после имени и фамилии, либо сообщить о незагруженной аватарке (см. скриншот задачи). Для получения имени, размера и временной папки для хранения загруженных файлов используйте суперглобальный массив $\_FILES. Для перемещения загруженной аватарки в целевую папку используйте функцию move\_uploaded\_file() (в качестве целевой создайте папку avatars). Чтобы правильно указать путь к вставляемой на страницу аватарке, не забудьте начать его с http://localhost/test/. При желании можете добавить проверку загружаемой аватарки на соответствие типу изображения и т. д.

Code: Code: <https://github.com/lyarov22/php-projects>/pract3



1. Создание и обработка форм

***Цель работы***: Научиться обрабатывать данные из формы. Заполнять таблицы БД данными, полученными из формы.

***Задание:*** Создать для своей БД форму и обработчик формы для заполнения двух таблиц, в которых не требуются данные из других таблиц (последнее будет темой следующего занятия )

***Отчетность:*** Загрузить архив папки с сайтом /OSPanel/domains/<имя папки>), а также \*.sql скрипт вашей БД, сделанный из PHPMyAdmin.

Рассмотрим комплексный пример обработки форм, в которой объединим обработку различных элементов html. Определим следующую форму:

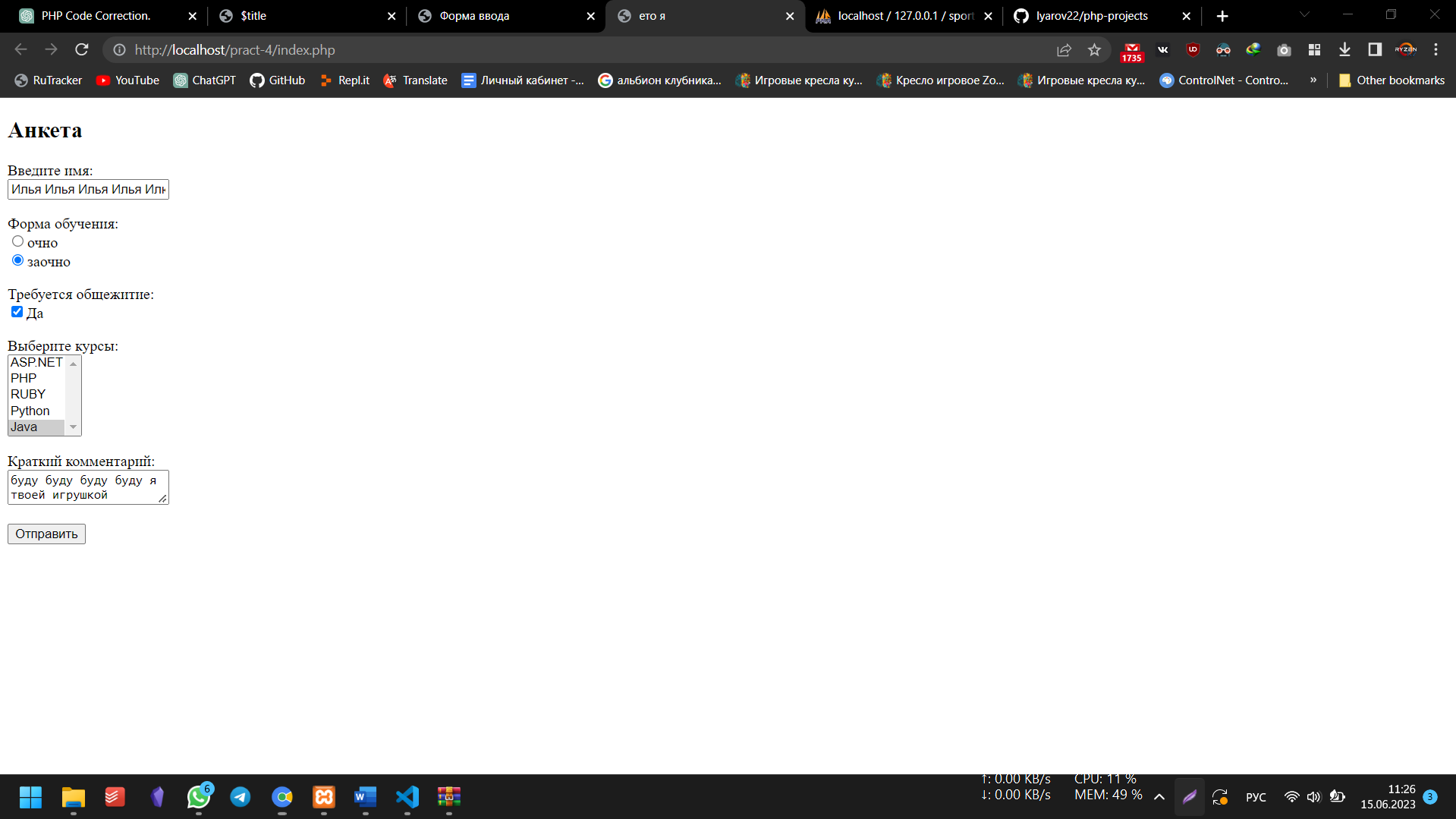
Здесь простейшая стандартная форма ввода анкетных данных:

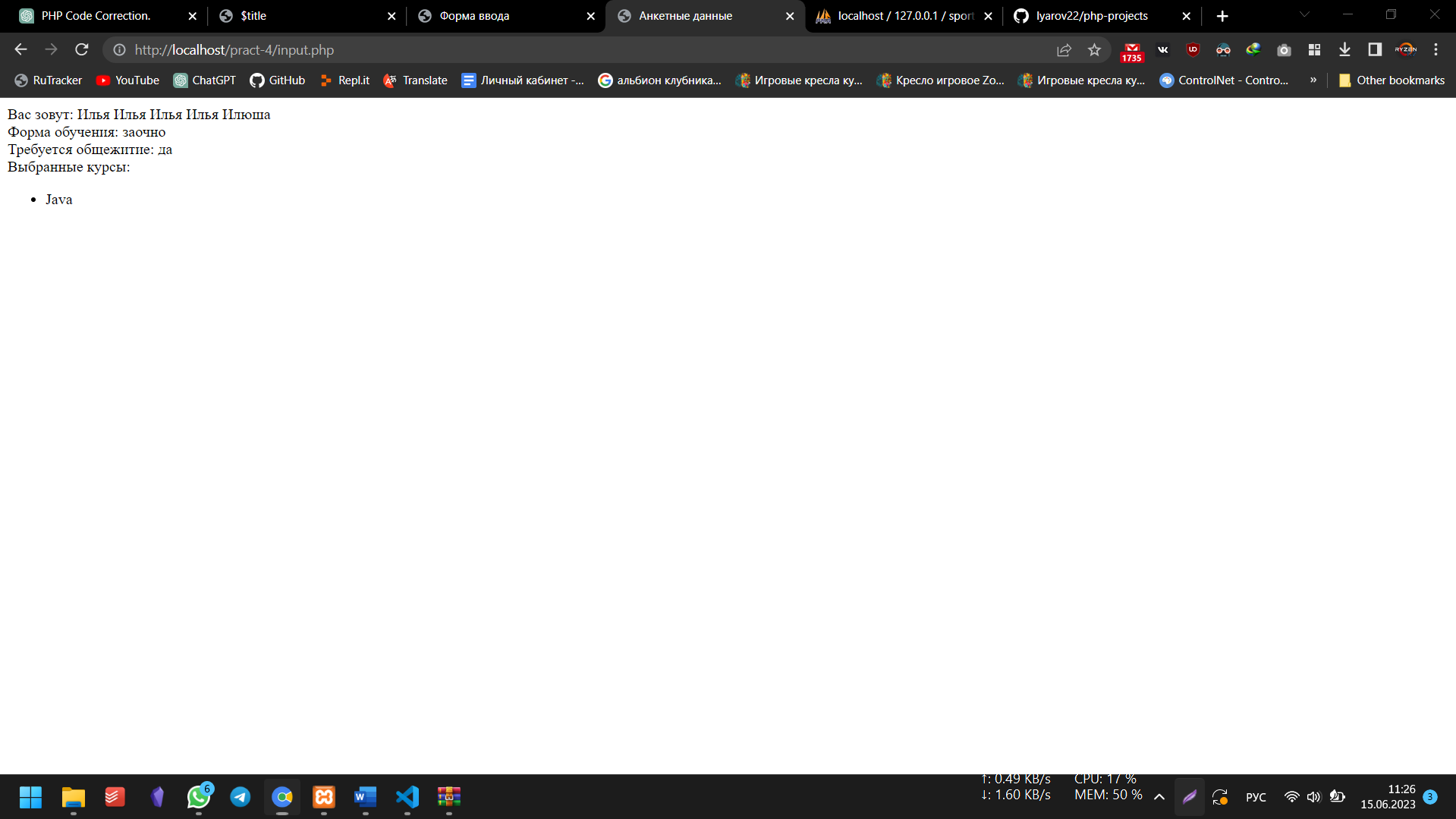
Теперь определим скрипт *input.php*, который будет обрабатывать эту форму:

oткроем страницу с формой, введем все данные:

И отправим форму на сервер:

Code: <https://github.com/lyarov22/php-projects>/pract4





1. **Заполнение таблиц БД данными, полученными из формы**

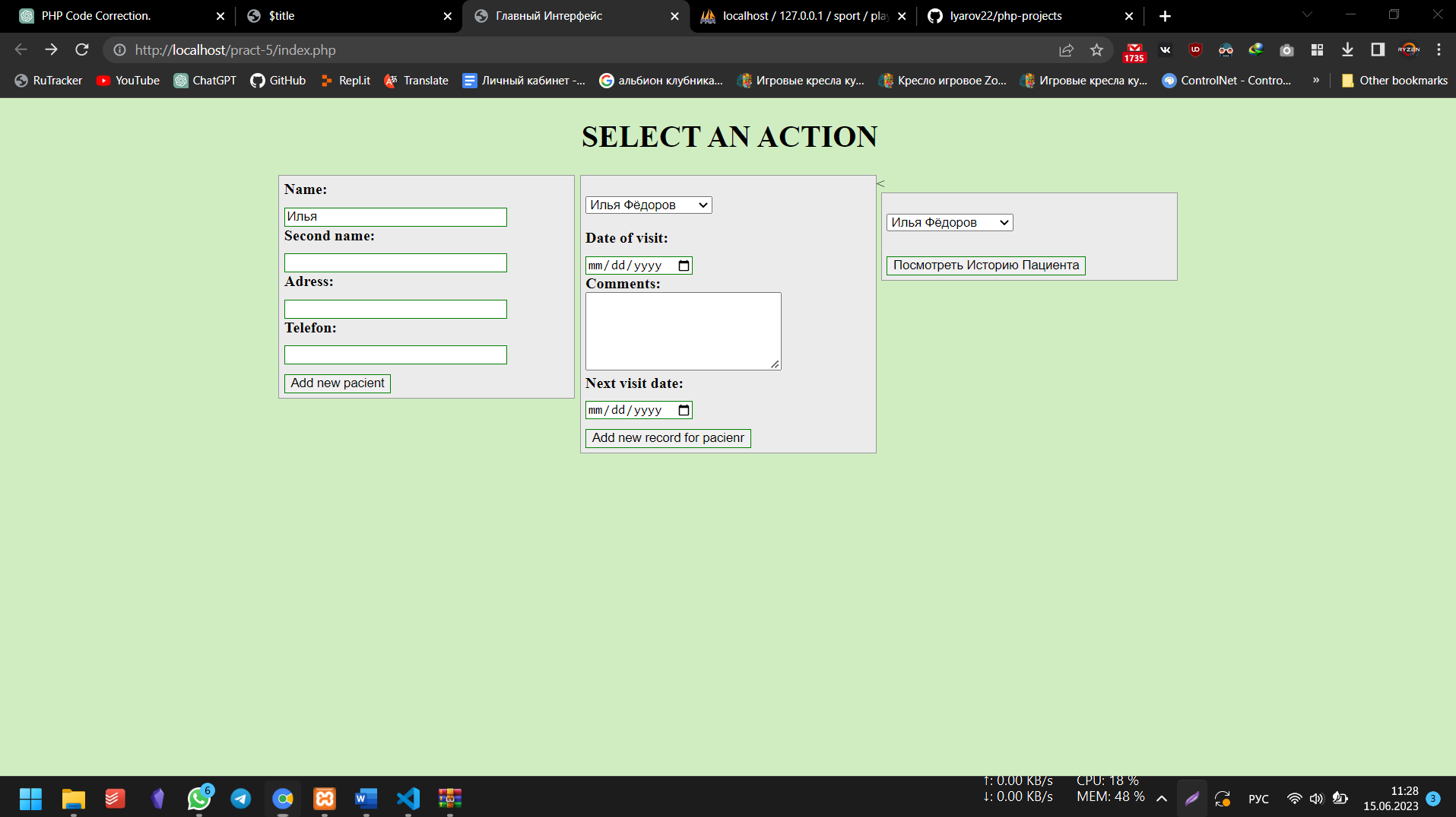
**Самостоятельная работа**

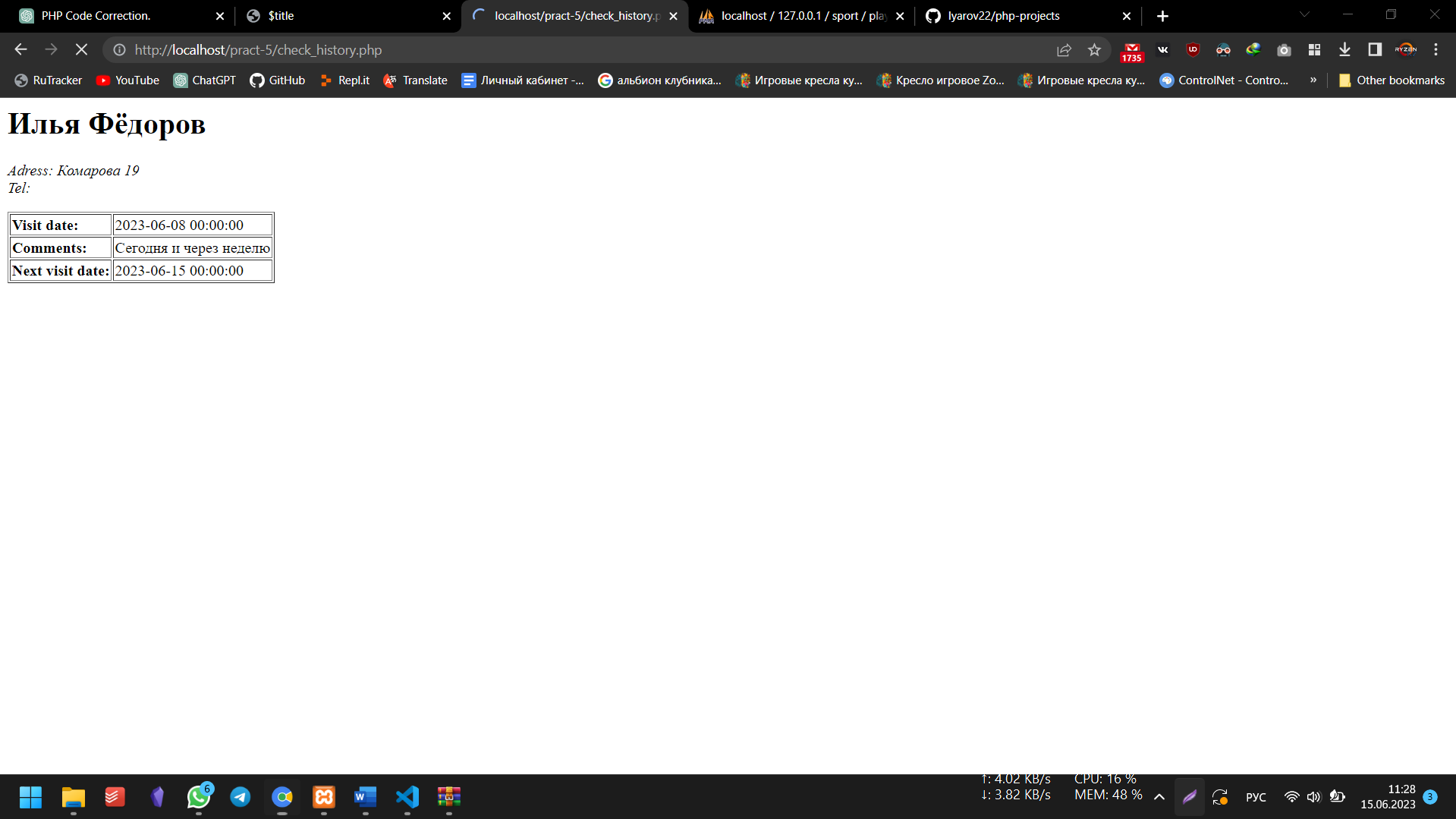
В первой части мы просто вывели данные, полученные через форму в виде суперглобального массива \_POST , в окне браузера. Теперь рассмотрим пример, когда эти данные заносятся в таблицу БД (test), таблица users поля id\_u, name, tel (id\_u - автозаполнение)

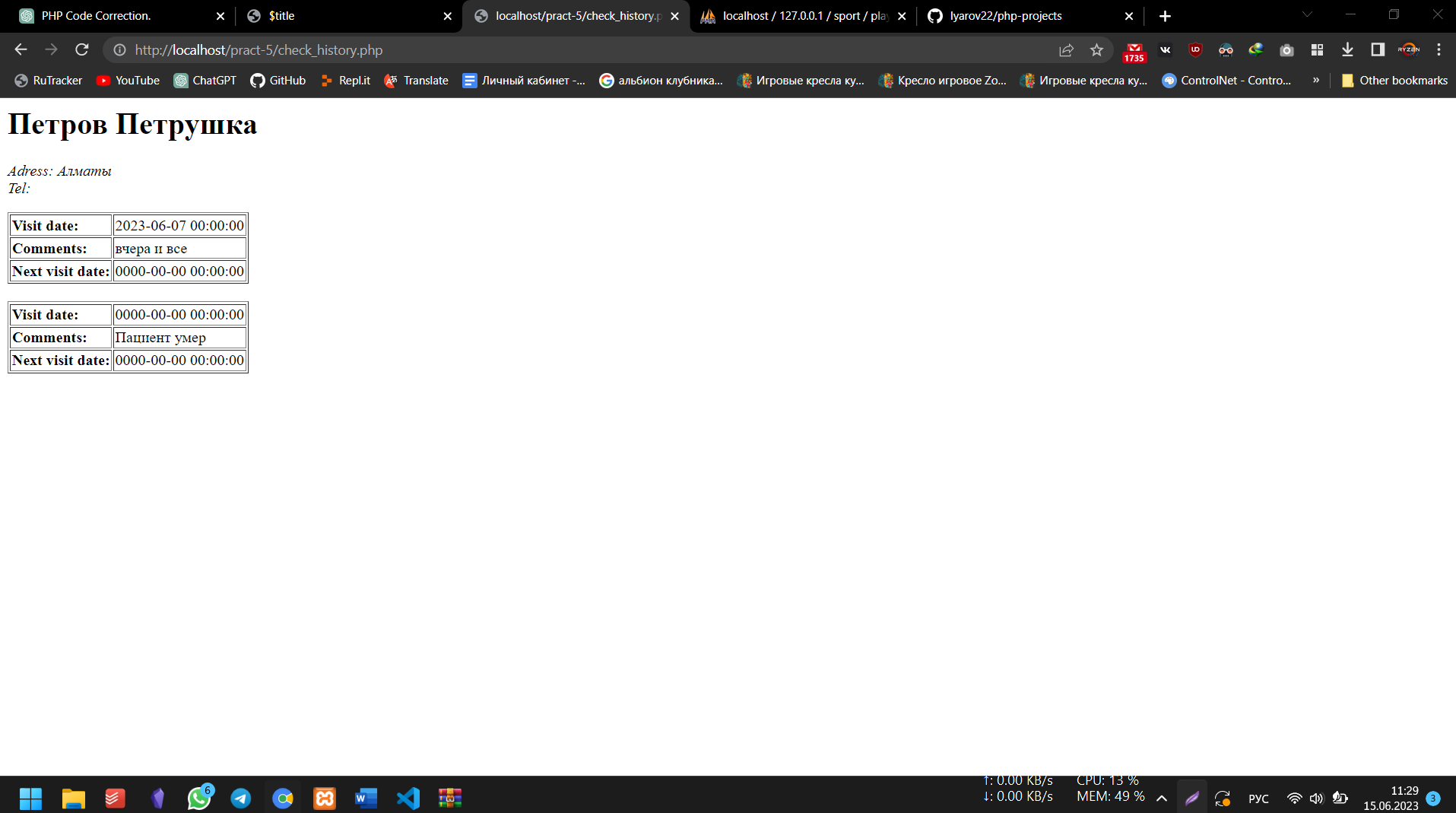
*Далее, при переходе по ссылке INSERT into table Users, попадаем в add.php*

*Далее, создаем обработчик add\_users.php*

Code: <https://github.com/lyarov22/php-projects>/pract5







6.**Создание простейшего веб приложения с использованием php + mysql**

**Самостоятельная работа**

В phpMyAdmin создаем базу данных sport

Создаем таблицы:

Таблица countries

country\_id country\_name

1 Россия

2 Англия

3 Испания

4 Италия

5 Германия

6 Франция

8 Португалия

Таблица players

player\_id team\_id country\_id player\_name

1. 1 1 Игорь Акинфеев

2. 2 2 Криштиану Роналду

3. 6 4 Марко Вератти

Таблица teams

team\_id country\_id team\_name

1 1 ЦСКА

2 2 Манчестер Юнайтед

4 4 Лацио

5 5 Бавария

6 6 ПСЖ

8 4 Ювентус

В папке domains создаем папку sport

В папке sport создаем файлы index.php, db.php, teams.php, players.php, countries.php, api.php, edit.php,

save.php

Code: <https://github.com/lyarov22/php-projects>/pract6

