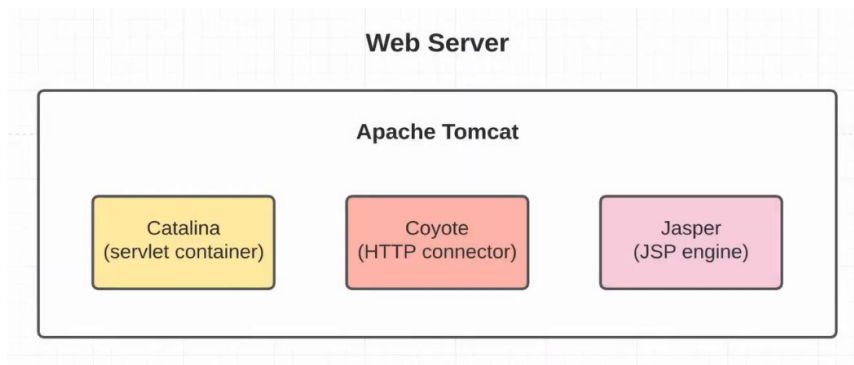
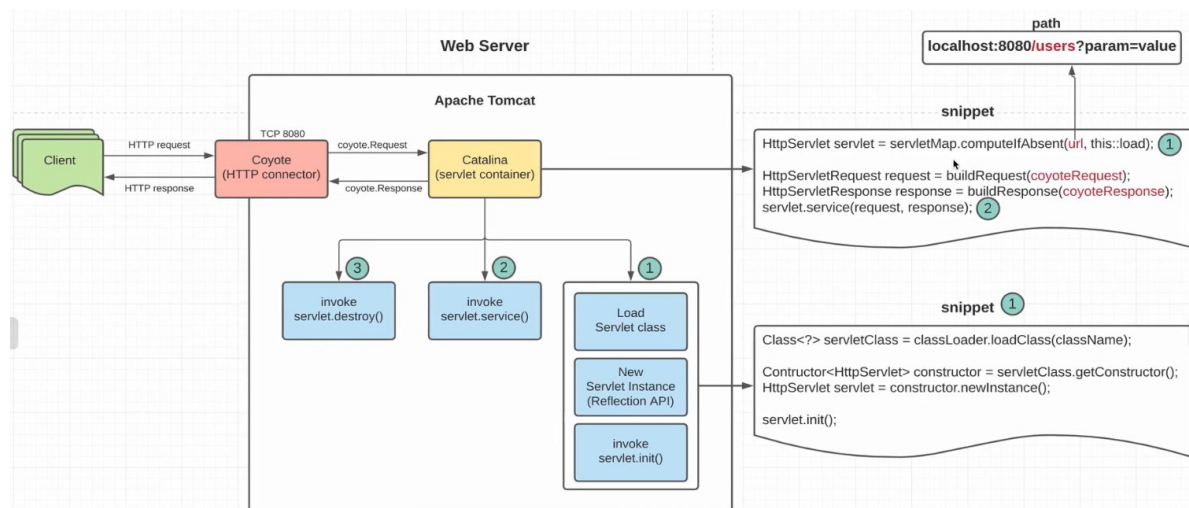


## Часть 1. Теоретическая.



### HTTP servers

[https://developer.mozilla.org/ru/docs/Learn/Common\\_questions/Web\\_mechanics/What\\_is\\_a\\_web\\_server](https://developer.mozilla.org/ru/docs/Learn/Common_questions/Web_mechanics/What_is_a_web_server)



### Сервлеты

<https://coderlessons.com/tutorials/java-tehnologii/uchit-servlety/servlety-kratkoe-rukovodstvo>

### Заголовки HTTP

<https://developer.mozilla.org/ru/docs/Web/HTTP/Headers>

### Тело запроса

<https://selectel.ru/blog/http-request/>

### CGI

<https://lectureswww.readthedocs.io/5.web.server/cgi.html>

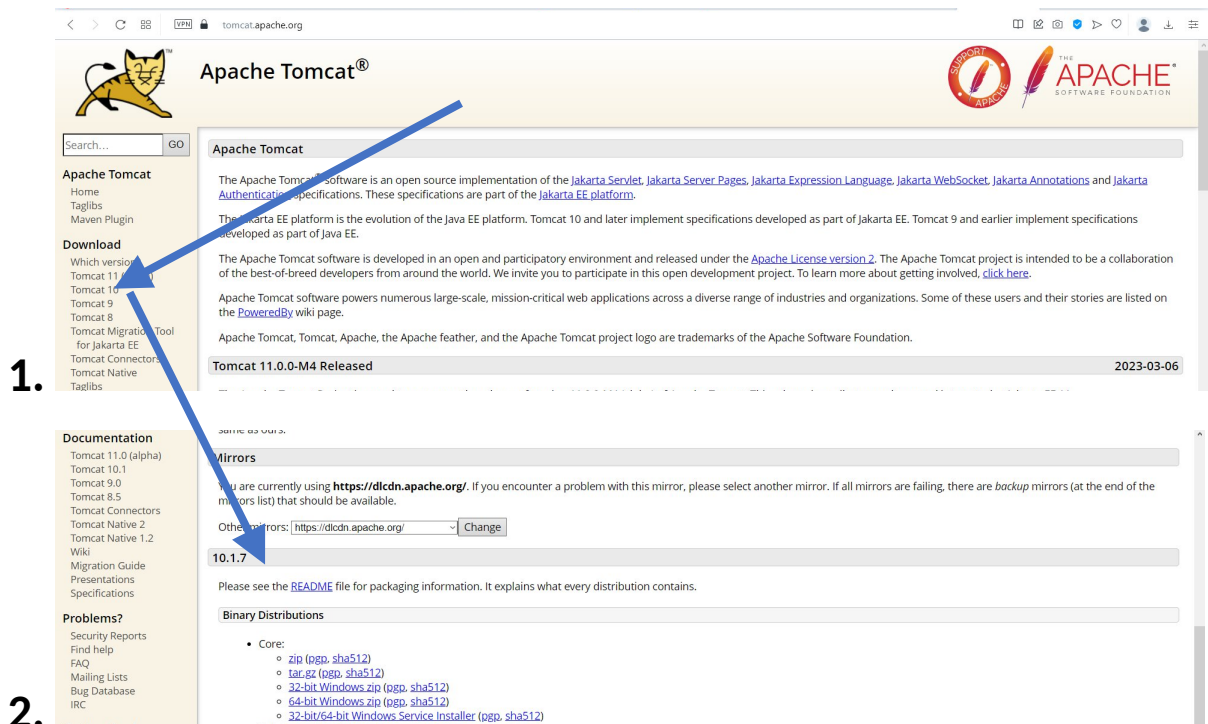
### Постман

<https://habr.com/ru/companies/maxilect/articles/596789/>

## Часть 2. Практическая.

### Установка веб- сервера.

Сначала необходимо установить веб сервер. Не обязательно устанавливать веб сервер заново. Если он уже установлен или имеется доступ к удалённому серверу можно использовать его.



### 3. После успешной установки и запуска вы можете наблюдать похожее сообщение:



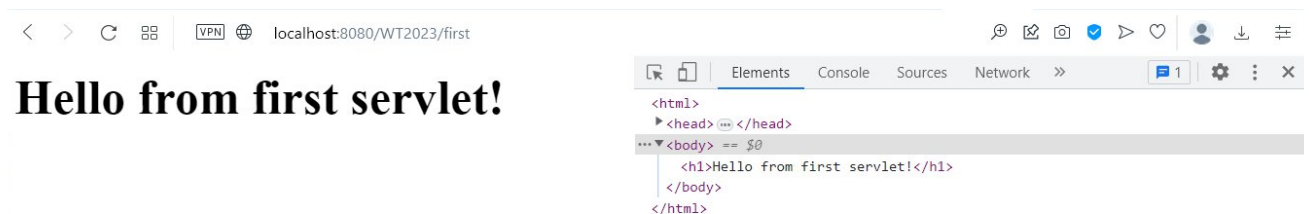
После установки веб сервера и понимания основ его работы, создадим простейший сервлет и намеренно допустим ошибку в тегах.

!(Важно) Если вы не пользуетесь автоматическим сборщиком проекта, не забудьте подключить файл `servlet-api.jar` из директории `apache-tomcat \ lib` к своему проекту.



```
1 package iit.mal.wt23.lab03.servlet;  
2  
3 import ...  
4  
5 @WebServlet("/first")  
6 public class FirstServlet extends HttpServlet {  
7  
8     @Override  
9     public void init(ServletConfig config) throws ServletException {  
10         super.init(config);  
11     }  
12  
13     @Override  
14     protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {  
15         resp.setContentType("text/html");  
16         try (var writer : PrintWriter = resp.getWriter()) {  
17             writer.write( s: "<h1>Hello from first servlet!</h2>");  
18         }  
19     }  
20  
21     @Override  
22     public void destroy() {  
23         super.destroy();  
24     }  
25 }  
26  
27  
28  
29  
30  
31  
32  
33 }
```

Записав свой сервлет в необходимые директории веб сервера и перезапустив сервер, вы сможете наблюдать подобную картинку, в которой браузер исправил нашу «ошибку» и привёл страничку к правильному структурному виду.

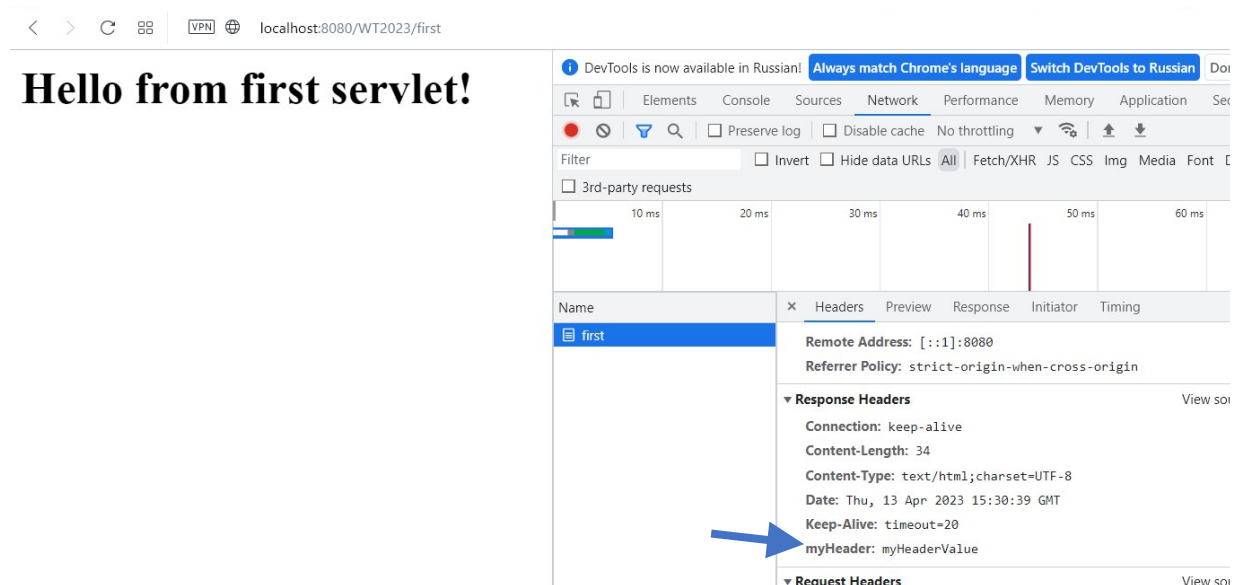


После этого, мы можем посмотреть выборочно те заголовки, которые нам интересны или вывести значение всех существующих. При этом мы можем задавать свои заголовки и устанавливать их значения.



```
1 package iit.mal.wt23.lab03.servlet;  
2  
3 import ...  
4  
5 @WebServlet("/first")  
6 public class FirstServlet extends HttpServlet {  
7  
8     @Override  
9     public void init(ServletConfig config) throws ServletException {  
10         super.init(config);  
11     }  
12  
13     @Override  
14     protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {  
15         req.getHeader( s: "user-agent");  
16         resp.setContentType("text/html");  
17         resp.setHeader( s: "myHeader", s1: "myHeaderValue");  
18         try (var writer : PrintWriter = resp.getWriter()) {  
19             writer.write( s: "<h1>Hello from first servlet!</h2>");  
20         }  
21     }  
22  
23     @Override  
24     public void destroy() {  
25         super.destroy();  
26     }  
27 }  
28  
29  
30  
31  
32  
33 }
```

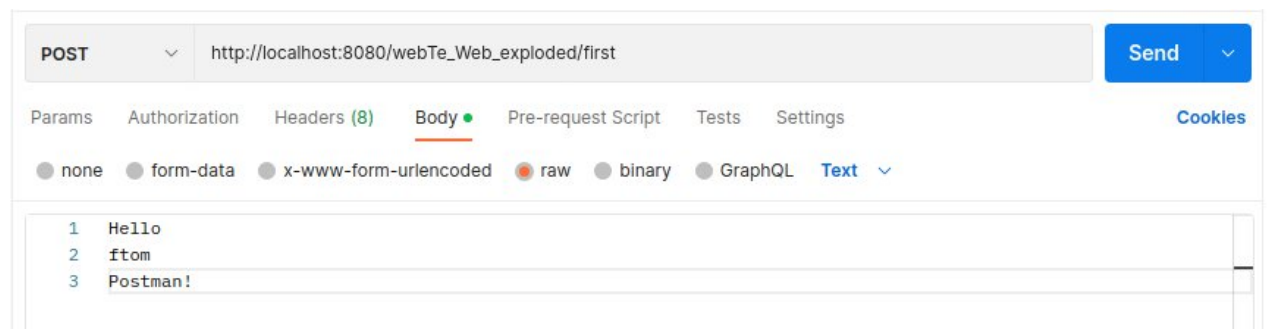
В браузере проверяем наличие собственного заголовка



Для передачи данных переопределяем метод `doPost` следующим образом:

```
@Override
protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
    try (var reader :BufferedReader = req.getReader()) {
        var lines :Stream<String> = reader.lines();
        lines.forEach(System.out::println);
    }
}
```

Установив Postman передаём в наше приложение следующие параметры:



Нажав кнопку Send получаем в приложении сообщение:

```
14-Apr-2023 10:11:09.365 INFO
14-Apr-2023 10:11:09.391 INFO
Hello
ftom
Postman!
```

А в Postman статус отправленного запроса и количество отправленных данных



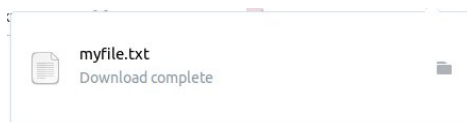
## Передача файлов:

Создадим новый сервлет Download:

```
import ...

@WebServlet("/download")
public class DownloadServlet extends HttpServlet {
    @Override
    protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
        resp.setHeader("Content-Disposition", "attachment; filename=\"myfile.txt\"");
        resp.setContentType("text/plain");
        resp.setCharacterEncoding(StandardCharsets.UTF_8.name());
        try (var writer :PrintWriter = resp.getWriter()) {
            writer.write("Servlet, hello from txt file!");
        }
    }
}
```

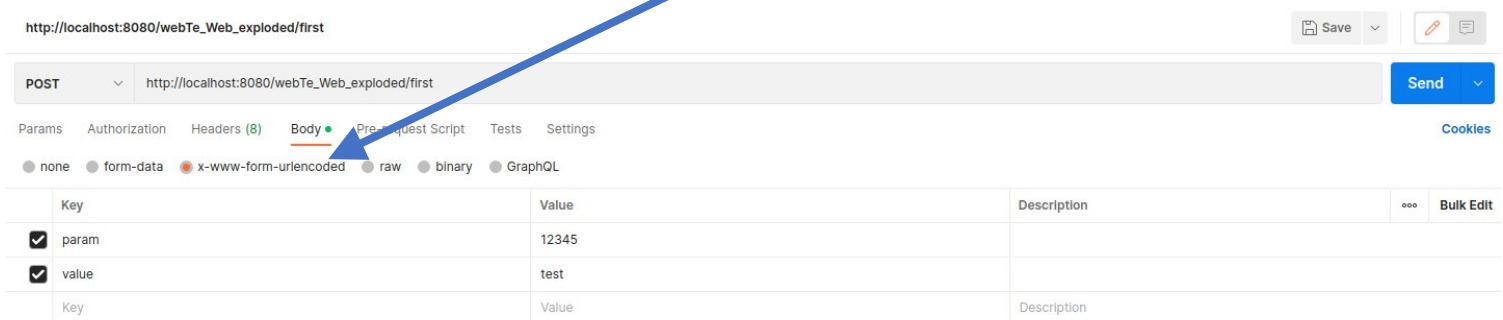
Произведя Redeploy приложения и указав этот сервлет в строке поиска браузера получаем автоматический вызов сохранения файла myfile.txt с текстом указанным нами в программе.



## Задание:

Необходимо разработать простейший сервлет, в котором:


- < Получить список всех заголовков и вывести на экран их значения;
- < Переопределить метод doPost для отображения переданных переменных в виде: param=1234&value=test  
(в Postman использовать эту вкладку)



- < Изменить сервлет работающий с передачей текстовых файлов для работы с бинарными, в частности отображения файлов изображений \*.jpg
- < Разместить файл с изображением в директории на жёстком диске, в браузере отобразить его с помощью изменённого сервлета.

### Часть 3. Контрольные вопросы.

По результатам выполнения лабораторной работы преподаватель надеется, что студент получил достаточные теоретические и практические знания и в состоянии ответить на следующие вопросы:

1. Что такое war-файл, его отличие от jar-файла?
2. Какова общая структура веб-проекта?
3. Что такое контейнер сервлета?
4. Каков жизненный цикл сервлета?
5. Что такое ServletContext, что он содержит?
6. Когда и сколько раз при (в какой фазе) выполнения сервлета вызывается метод `init()`?
7. Что такое параметры запроса, какой у них тип и как передать массив параметров по одному ключу?
8. В чём отличия значений передаваемых параметров в опциях Postman?  

9. Какой метод в HTTP не является неизменяемым?
10. Что такое CGI отличие этой технологии от сервлетов?

Для защиты лабораторной работы необходимо:

- ⟨ Прочитать и понять теоретическую часть.
- ⟨ **Индивидуально** выполнить все задания (теоретические и практические) согласно варианта - если такой присутствует. Принимается первый уникальный код, все остальные признаются плагиатом или репликами первого и не учитываются.
- ⟨ Понимать, что делает каждая строка написанного кода.
- ⟨ Оформить отчёт, который должен быть распечатан или написан вручную на листах формата А4 (цвет листов не важен) и содержать титульный лист и ход выполнения работы. Выводы о результатах проделанной работы приветствуются.  
\* использование репозитория (git или аналоги) с открытым доступом приветствуется.

*P.S. Для тех, кто не имеет доступа в сеть Интернет, все программы необходимые для выполнения данной лабораторной работы находятся в папке с заданием.*