com.microsoft.sqlserver:mssql-jdbc:10.2.1.jre11

jakarta.servlet.jsp:jakarta.servlet.jsp-api:3.1.1

jstl:jstl:1.2

log4j:log4j:1.2.17

1. JDBC

FromRaR

Используя драйвер JDBC выполнить организацию соединения с БД.  
Она должна содержать author, book\_name, publication\_year, pages.  
Написать страницу JSP с кнопкой “NEXT”, по нажатию на которую  
будет выводиться последовательно по две записи (в виде таблицы с заголовками) из бд.  
Используйте maven для сборки.

2. JDBC

task2

Используя драйвер JDBC выполнить организацию соединений с базой данных.  
БД должны содержать text(сообщение) и like и dislike(целое,счетчики).  
На первой JSP страниуе разместите кнопку GO.  
По нажатию на которую выполняется переход на страницу look.jsp с текстовым сообщением,  
количеством like и dislike и двумя кнопками «Like» и «Dislike» под ними,  
которые увеличивают значения в счетчике.  
Используйте maven для сборки.

3. JDBC

* ServletLCS(не работает)

Используя драйвер JDBC выполнить организацию соединения с БД.  
БД должна содержать login и sum (sum - числово, сумма денег на счету).  
На первой JSP странице разместите поле ввода login и кнопку “Sign in”.  
В случае соответствия значения login со значением в БД  
выполнить переход на страницу work.jsp с тремя кнопками и полем ввода.  
“GETINFO” - выводит сумму на счету (sum).  
“PAY” - отнимает введенное значение от sum, если это возможно (сумма на счету не может быть отрицательной),  
“ADD” - зачисляет введенную сумму на счет.  
Все действия с sum записываются в лог файл.  
Если login введен неверно, перейдите на страницу error.jsp с соответствующим сообщением и ссылкой на страницу входа.

4. Сокеты

YesNo (выводит статистику только для 1 клиента)

Создать клиент-серверное многопоточное приложение на сокетах.

Клиенты посылают серверу три типа сообщений:/no /yes /getall /help.

После этого сервер подсчитывает количество /no и /yes

посланными клиентами (клиент может послать одно и то же сообщение несколько раз),

на /getall выводит статистику, на /help список доступных команд.

На все другие сообщения сервер не реагирует.

Продемонстрируйте работу для трех клиентов и одного сервера.

6. Теги

*zadanie\_jstl*

Создать тег-функцию для проверки корректности номера договора.

Корректный номер содержит от 4 до 6 цифр,

потом /21- и 4 символа латинского алфавита в верхнем регистре (123456/21-ABCD - корректно, 12/21-AAAA - некорректно).

Создать jsp страницу с полем для ввода текста и кнопкой check.

Результат проверки можно вывести на той же странице или создать еще одну jsp страницу

7. Сервлет JSP

last

Создать сервлет и взаимодействующие с ним классы и JSP- страницы, выполняющие следующие действия.

Пользователю выводится страницы с двумя кнопками GET и POST для выполнения get и post запроса соответственно.

При нажатии на кнопку информация о типе и времени запроса заносится в файл cookie

и при повторном переходе выводится на стартовую страницу.

По нажатию на кнопку GET выведите: время последнего get запроса, название и версию протокола.

По нажатию на кнопку POST полностью (с заголовками) содержимое HTTP-ответа.

8. Сервлет JSP

Есть два чекбокса(user admin) на странице и поле для ввода пароля.

Написать сервлет, который при отправке формы, проверяет соответствует ли пароль правилам:

от 6 до 12 символов, содержит цифры и символы.

Если пароль верен, то в зависимости от чекбокса показываем страницы.

Если user - страница user, если admin - записать в куки текущее время, вывести приветствие админа и время прошлого захода

9. Куки

cook

СО СЛОВ СДАЮЩЕГО

Там короче надо было пользователю ввести о себе группу курс специальность и т.д.,

это все сохраняется в куки и пользователя перебрасывает на главную страницу.

При попытке повторной авторизации введённые данные проверяются с куки и если неверные,

то выкидывает ошибку и перебрасывает на авторизацию опять.

Ещё добавить кнопку удаления, чтобы пользователь мог удалить свои данные с куки

10. Сокеты

* first5

СО СЛОВ СДАЮЩЕГО

Создать клиент серверное многопоточное приложение на сокетах.

Клиенты посылают серверу любые сообщения, после этого сервер посылает каждому клиенту целое число,

соответствующее порядковому номеру подключившегося клиента.

Первому на все сообщения посылается 1 второму - 2.

Продемонстрируйте работу для трех клиентов и одного сервера.

* Сделать сервлет и два фильтра. Первый фильтр логирует дату, порт, хост, локаль пользователя. Второй фильтр проверяет введённое число, если меньше 7777, вывести OK 200. Если ни один из фильтров не отработал (я сделал если не отработал 2) вывести 404 Not Found
* servlet1
* Разработать сокетное приложение. Клиент пишет в консоль команду /topic и ему выводятся все темы, которые хранятся на сервере в виде статической Map коллекции(это коллекция ключ-значение, ключи были топики, а значения просто текст к этому топику, 2 стринга, ниче сложного). После того, как клиенту вывелись топики, он может написать любой из топиков, и ему выведутся все сообщения с этим топиком(все значения кол-ции) map. Показать работу приложения запустив сервер и двух клиентов
* Этого нет
* используя дрвайвер jdbc выполнить организацию соед с бд.бд должна содержать login,code,sum.на первой Jsp старнице разместите поля ввода login,code и кнопку login.в случае совпадения со значкние в бд выполнить переход на страницу action.jsp с двумя кнопками check-выводит сумму на счету (sum),"pay"-отнимает 100 от sum,если возможно .все действия с sum запис в лог файл. если login и code введен не верно перейдите на страницу с соотв сообщением
* JSP и сервлет, два пользователя(admin, user). Получить время создания сессии, ид сессии, лог файл
* ServletLCS(не работает)
* используя драйвер выполн опер с бд  
  name email phone  
  кнопка get  
  и вывести в табл виде name phone
* first4
* создать клиент серверное многопоточное приложение на сокетах.клиенты посылают серверу любые сообщения. после этого сервер посылает каждому клиенту целое число соотв порядковому номеру подключившегося клиента. Нр первому на все сообщения посылается 1 второму 2.продемонстрируйте работу для трех клиентов и одного сервера
* first5
* Cоздайте сервлет и два фильтра. Первый пишет в лог время и ip клиента а второй проверяет что введены  
  телефон и мыло и перенаправляет на страницу приветствия, если оба значения отсутствуют добавьте сообщение об ошибке
* 2filter
* Использовать сервлет/jsp.Есть 4 поля для ввода переменных a, b, c, x. Если какое-то из них пустое, то в новой странице выводится сообщение об ошибке. Если есть все значения, то считается на новой странице y=a+ bx+cx'2.
* Servlet\_ABCX
* last2(тут что-то не работает)
* Создать сервлет и взаимодействующие с ним классы и JSP-страницы, выполняющие следующие действия. Пользователю выводится страница с двумя кнопками GET и POST для выполнения get и post запроса соотв. При нажатии на копку информация о типе запроса заносится в файл coocike и при повторном переходе выводится на стартовую страницу. По нажатию на GET выведите название и версию протокола, MIME тип запроса, определение имени польователя, выполнившего запрос. По нажатию на кнопку PUT всю информацию из заголовка запроса
* ServletGPP(не оч)
* last(выглядит пизже)

Перенаправление на другую страницу

String path = "/index.html";

ServletContext servletContext = getServletContext();

RequestDispatcher requestDispatcher = servletContext.getRequestDispatcher(path);

requestDispatcher.forward(request, response);

переадресация

String path = request.getContextPath() + "/notfound";

response.sendRedirect(path);

request.getRequestDispatcher("next.jsp").forward(request, response);