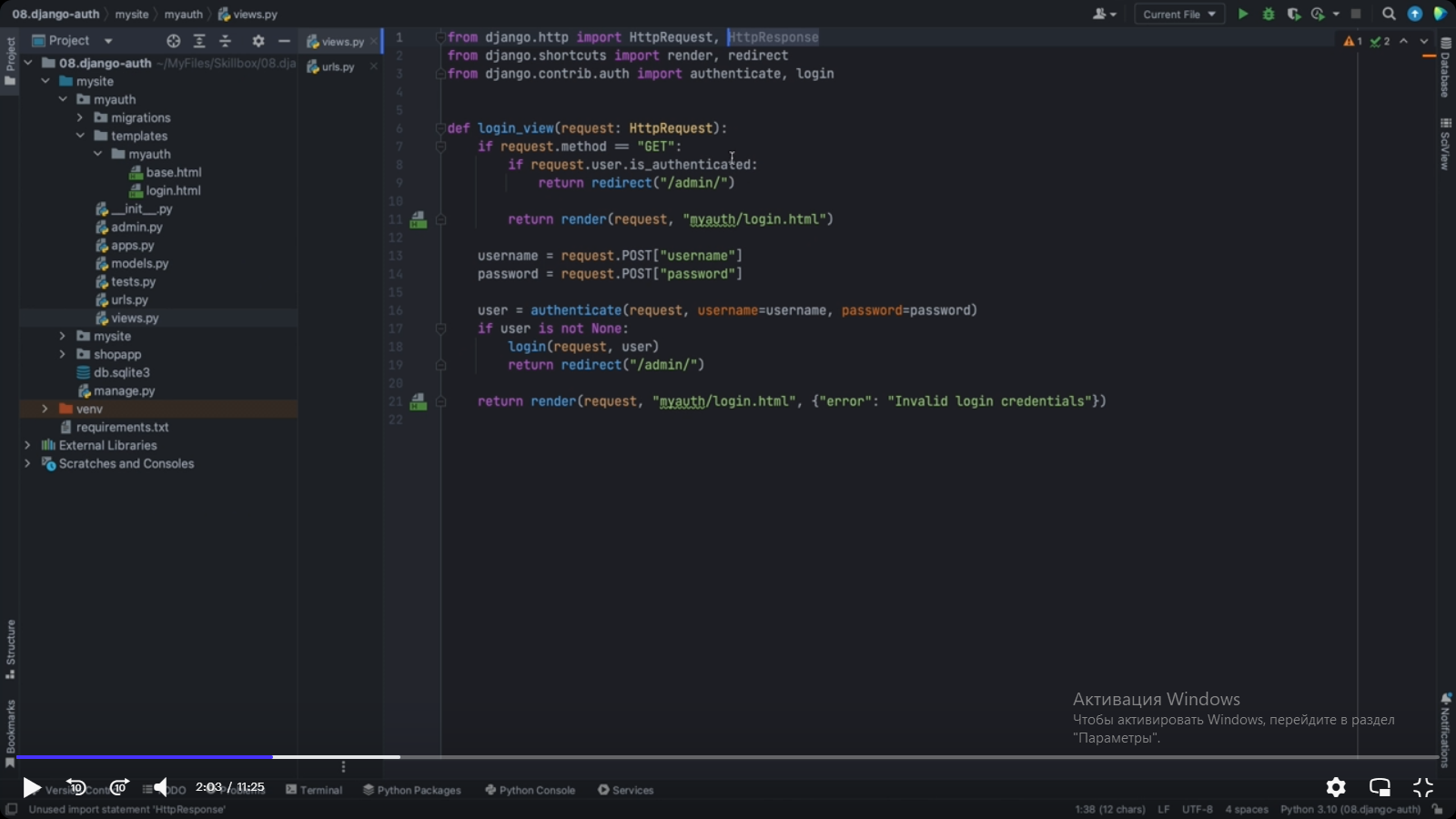
**8.4 Пользователи и сессии, куки**

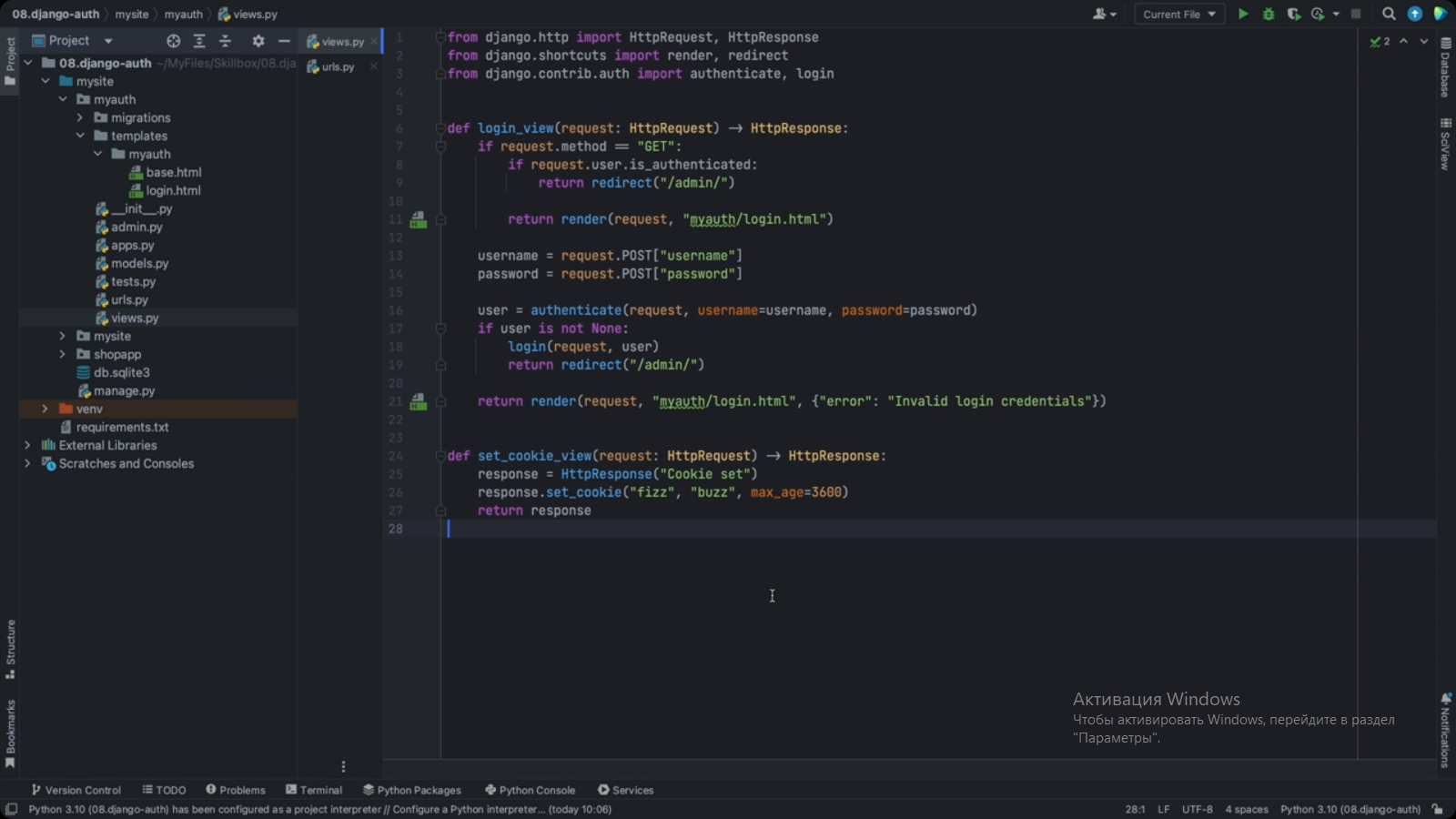
Некоторые пользовательские данные могут храниться в браузере. Для этого сервер передает браузеру информацию что можно сохранить. Для этого используются куки – простые текстовые файлы. Эти файлы хранятся в браузере и передаются с каждым запросом для того чтобы сервер понимал с какими данными пользователь сейчас работает. В куках может храниться любая информация какую вэб сайт может сохранить. К примеру, можно хранить информацию какой язык выбрал пользователь или какие у него есть предпочтения или историю посещения. Очень часто куки используются для отслеживания действий пользователя и контекстной рекламы. Куки обычно сгорают по истечении некоторого времени, которое задаётся в их настройках. Эти настройки присылает сервер браузеру. Если же сервер присылает браузеру срок жизни 0, то куки сгорают сразу же после закрытия браузера. Доступ к кукам можно ограничить, сделав их htp only. В таком случае djava\_script, который работает в браузере не сможет получить доступ к этим данным и куки будут переданы с обычным запросом на сервер, где их можно будет прочитать. Такое ограничение снижает опасность того, что куки будут украдены злоумышленником, который внедрит свой java\_script на сайт. Однако всё равно в куках нельзя хранить какую-либо чувствительную информацию, т.к. она может быть не только украдена, но и утеряна, например, если пользователь почистит кэш. Поэтому стоит использовать куки как временное хранилище, например, чтобы сохранить там информацию об аутентификации пользователя. Соответственно если он почистит куки, то ему нужно просто повторно войти, кука будет установлена в браузере и он снова сможет пользоваться сайтом.

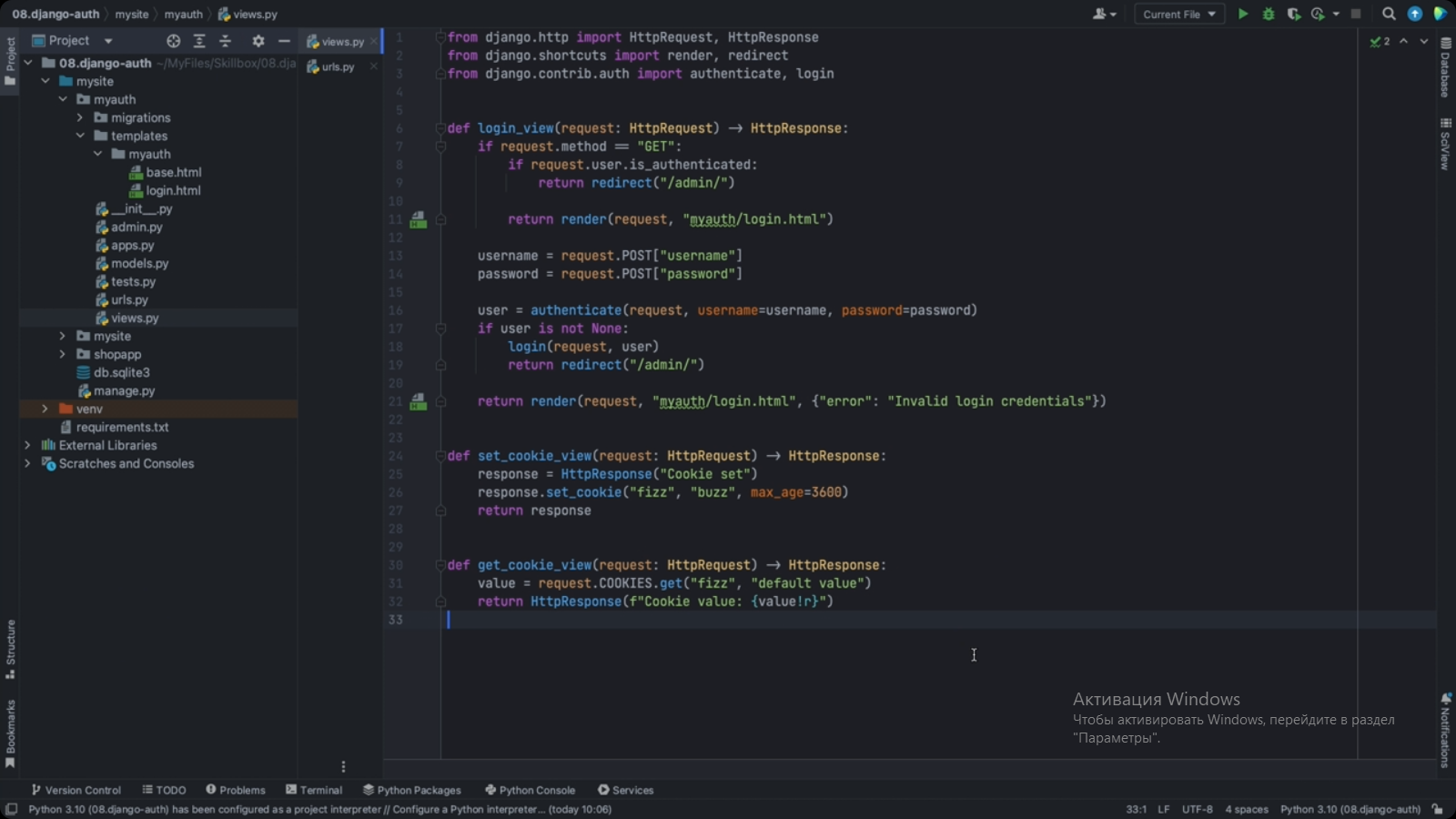
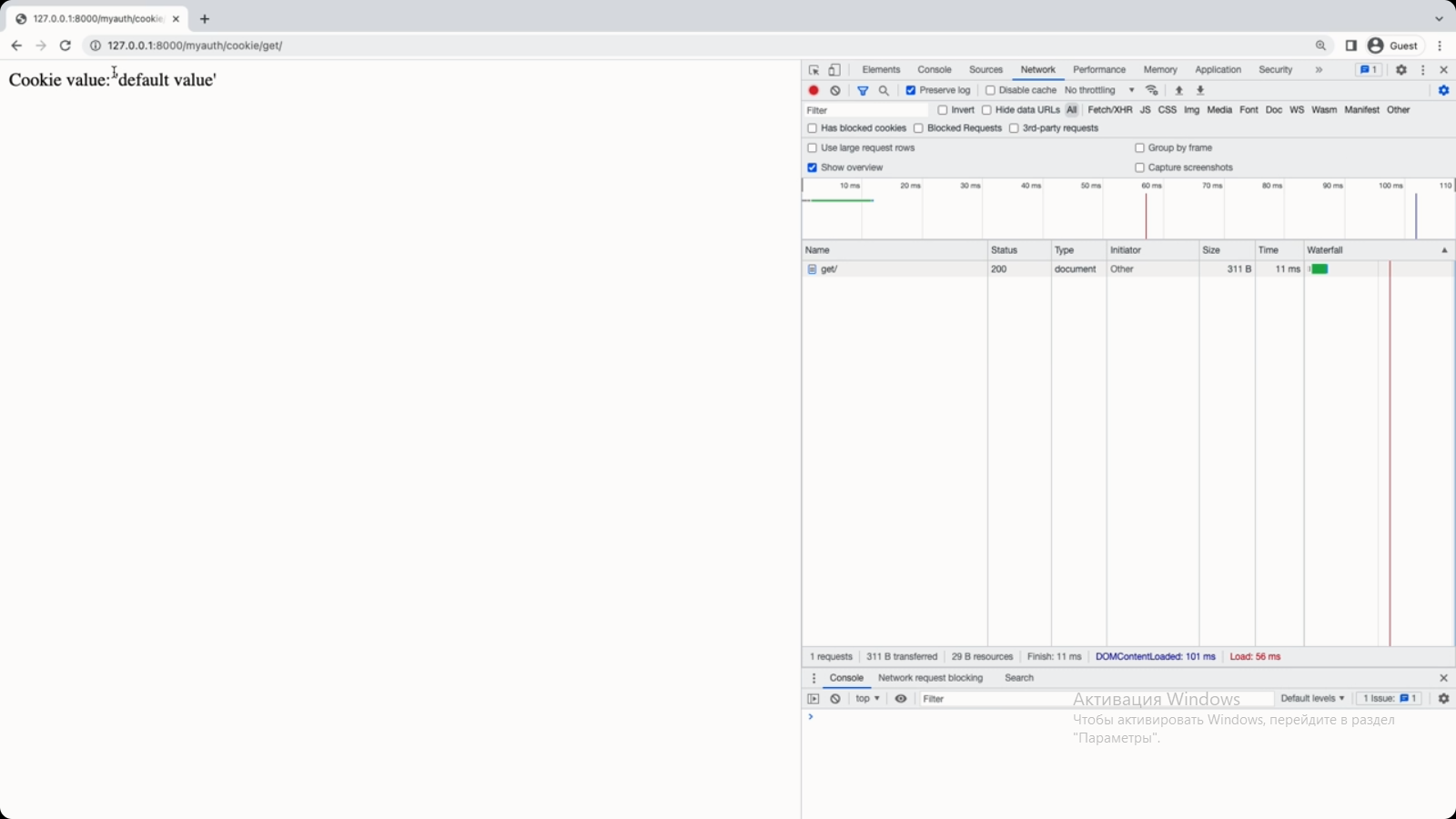
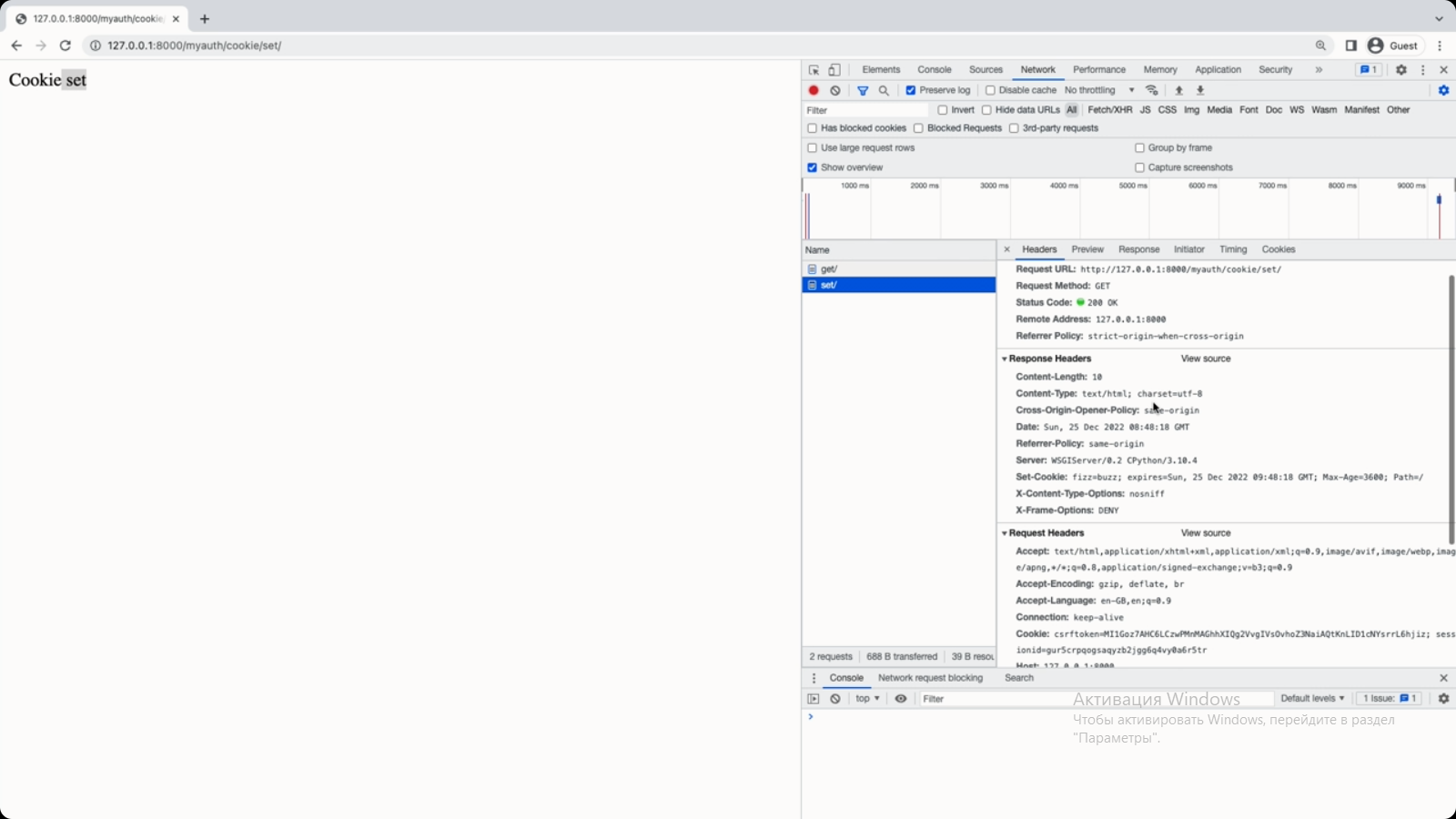
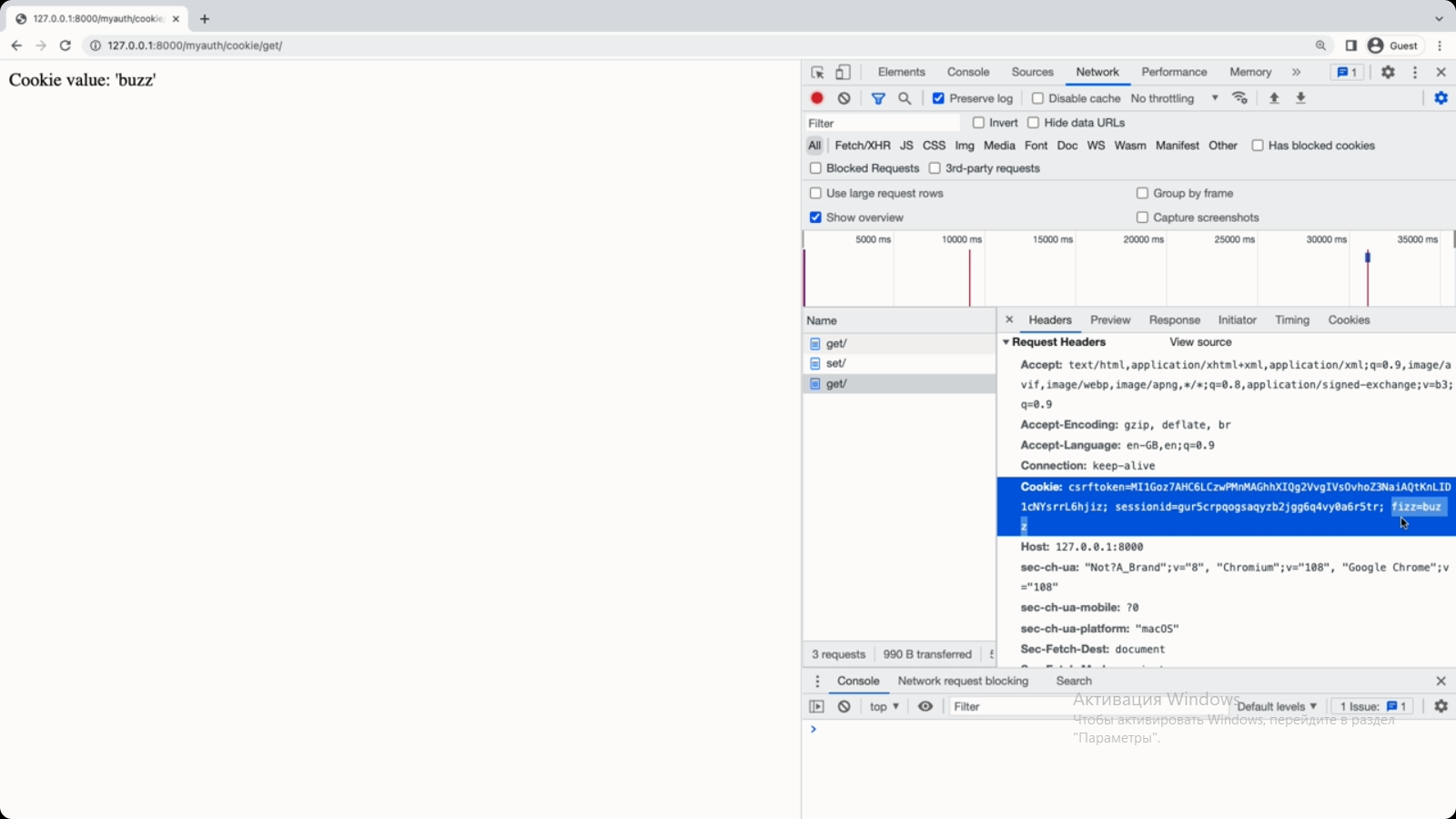
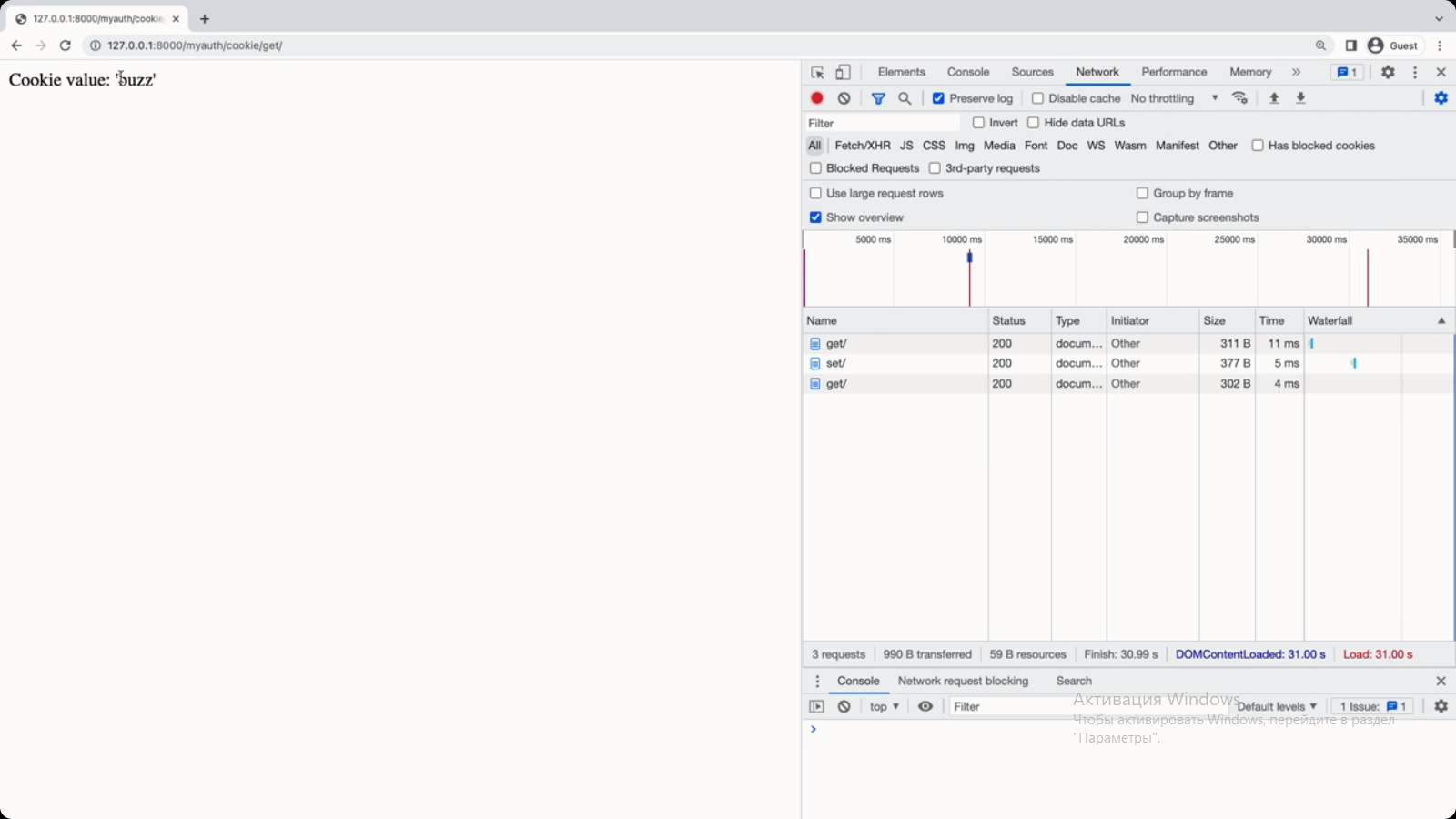
Также нужно быть осторожным с хранением чувствительной информации в браузере, т.к. это может привести к отслеживанию пользователя или к нарушению конфиденциальности.

В Django куки управляются при помощи менеджера куки, который позволяет удобно создавать и удалять куки, а также устанавливать их срок жизни и прочие настройки.

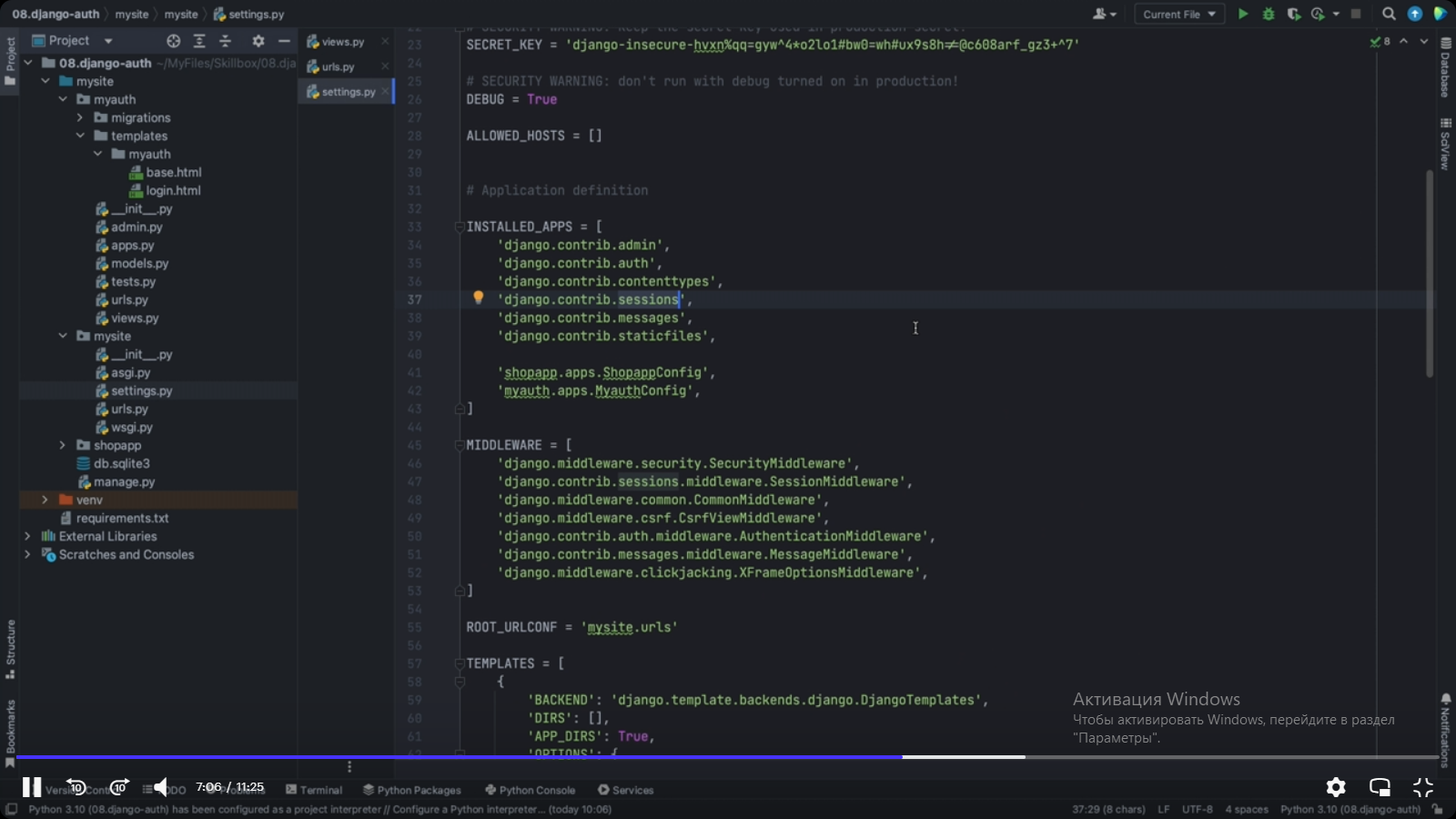
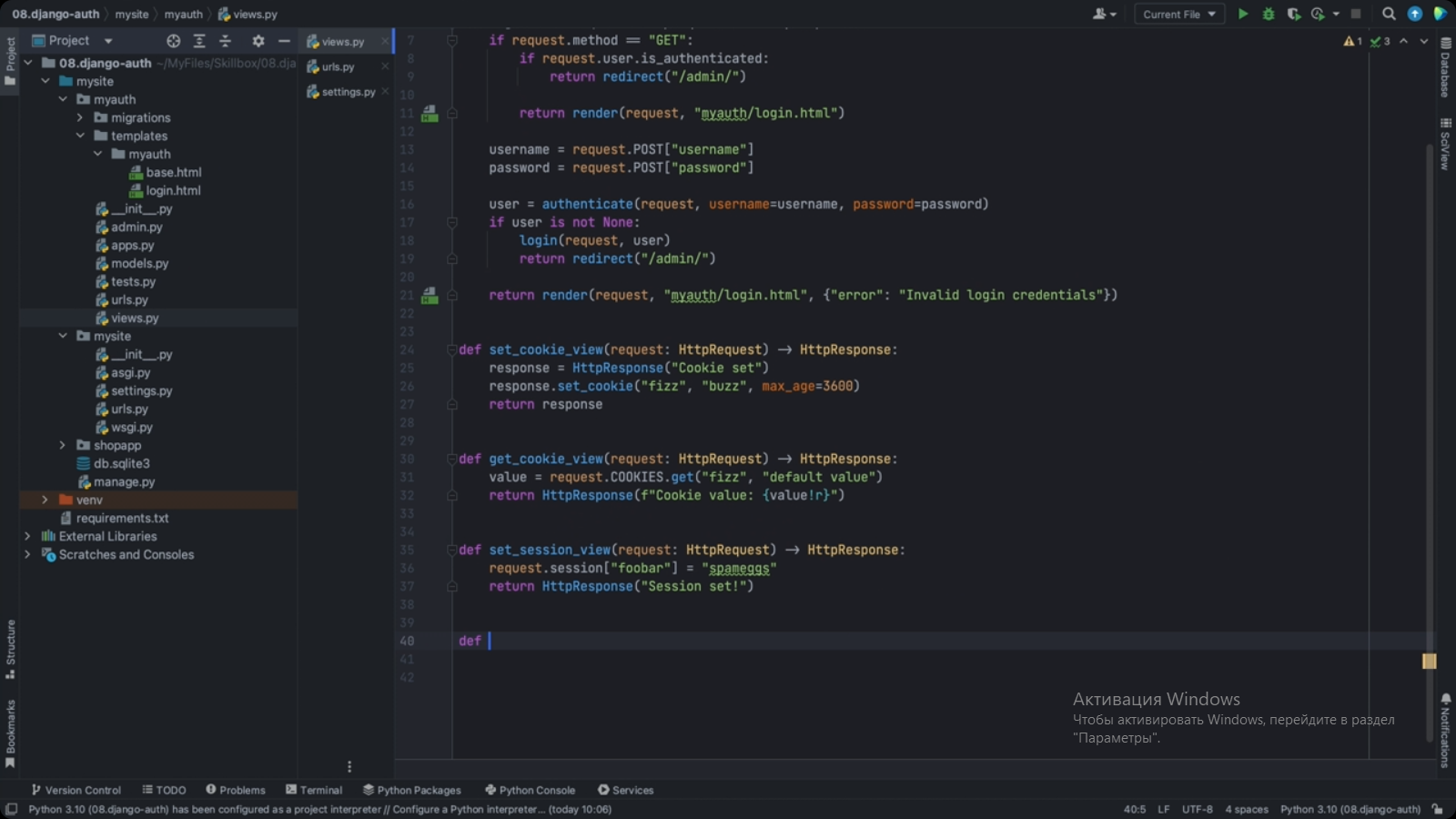
Чтобы записать куки мы можем использовать метод set\_cookie на классе HttpResponse, поэтому импортируем его

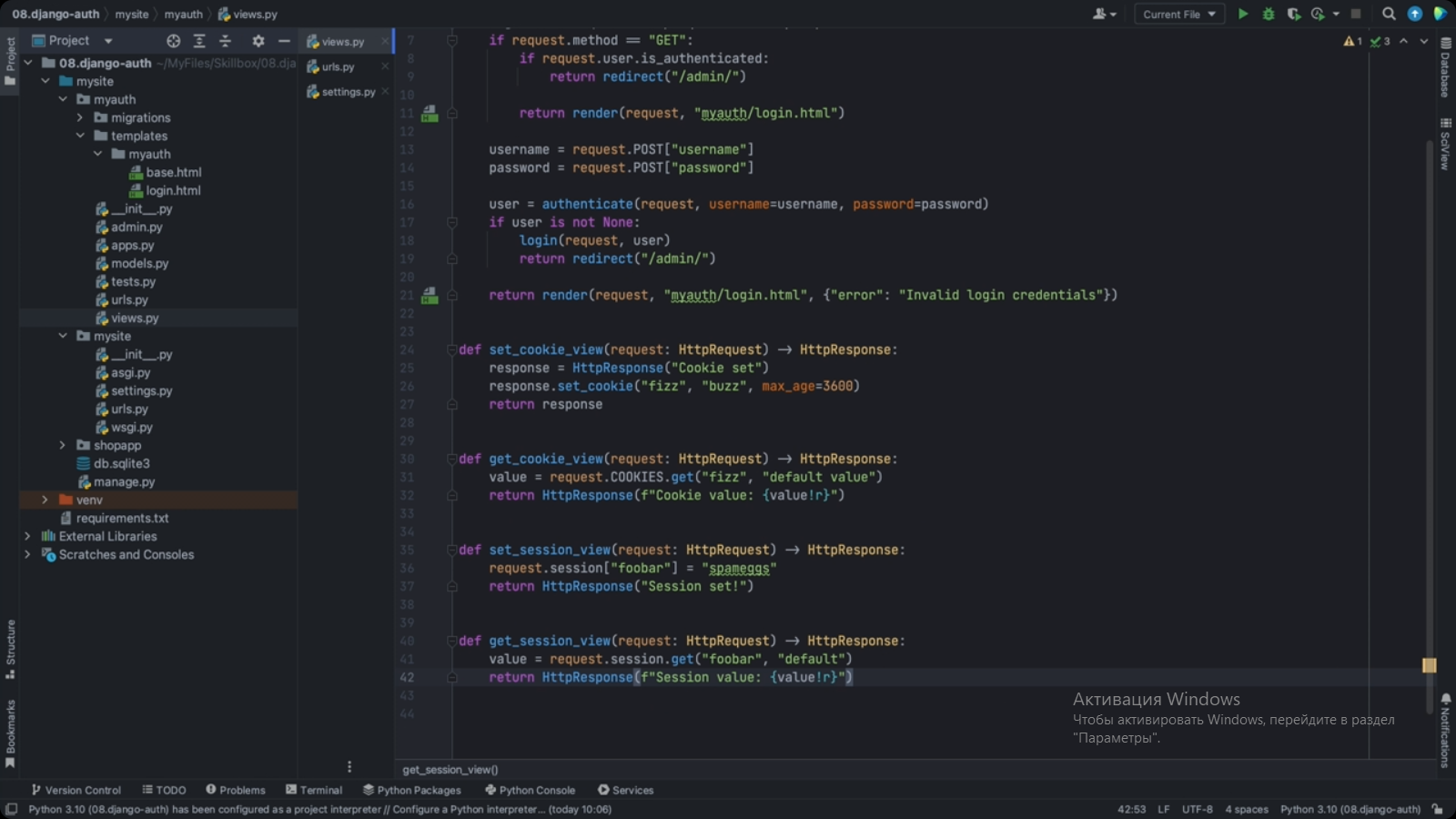
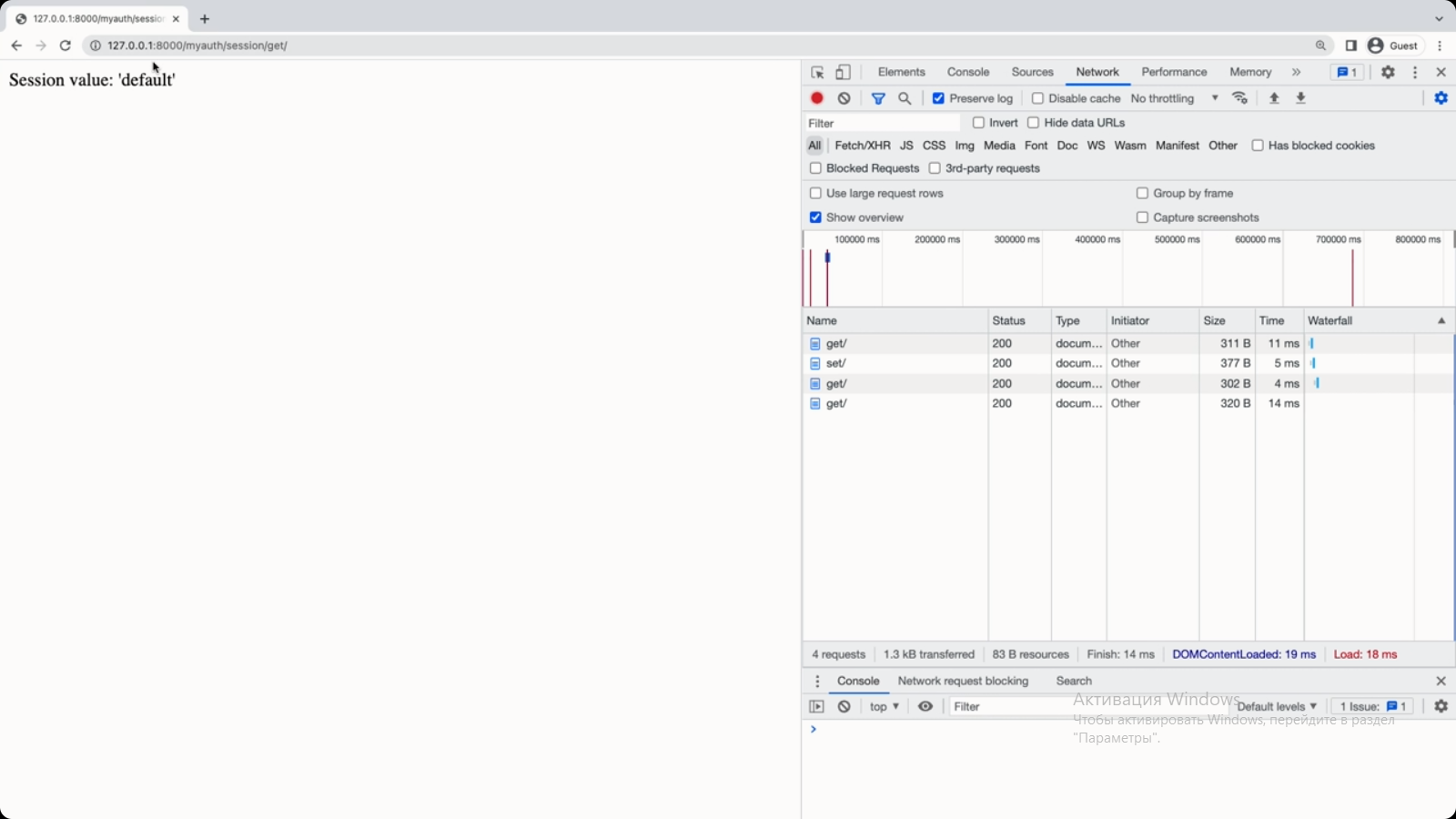
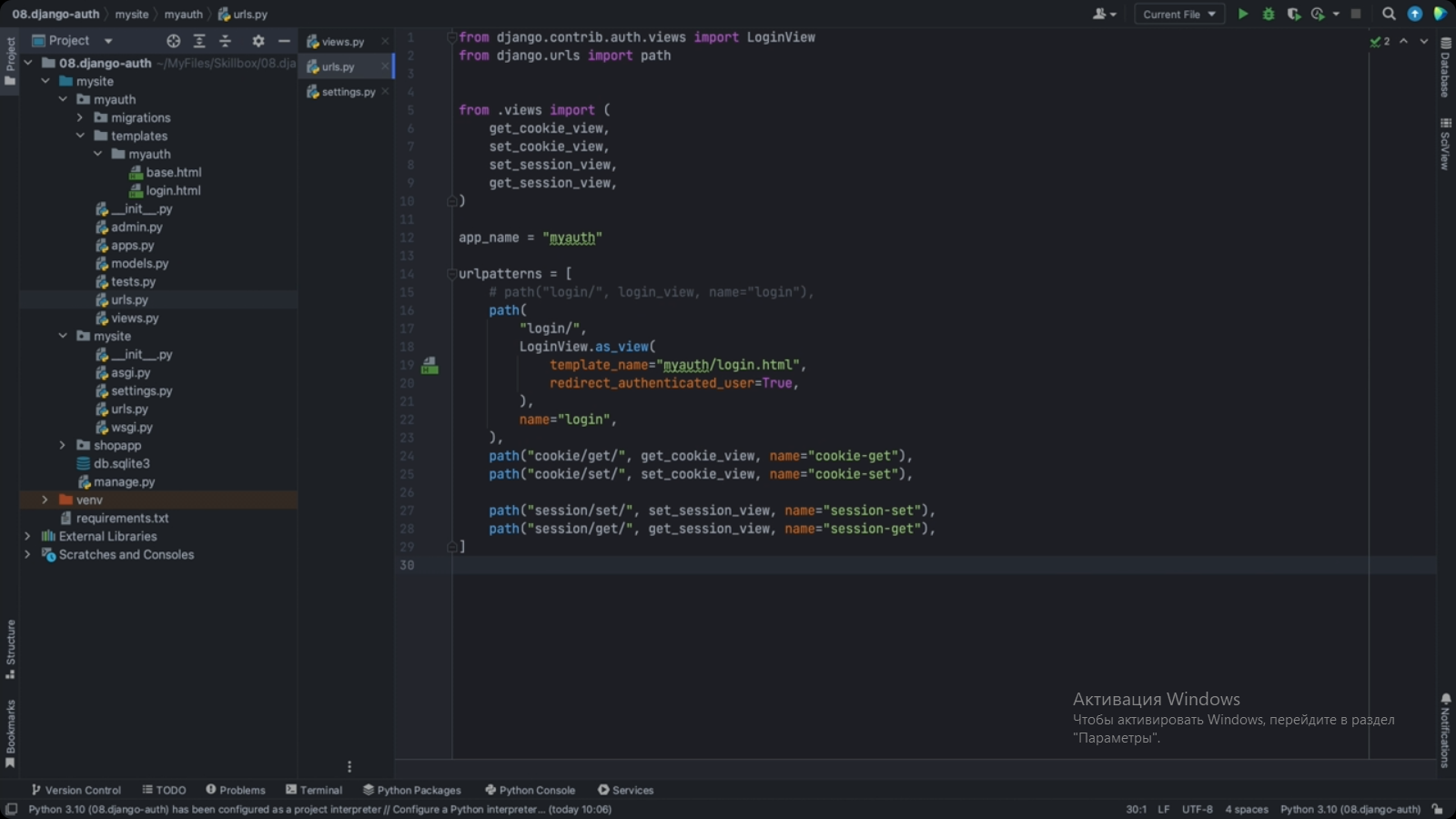
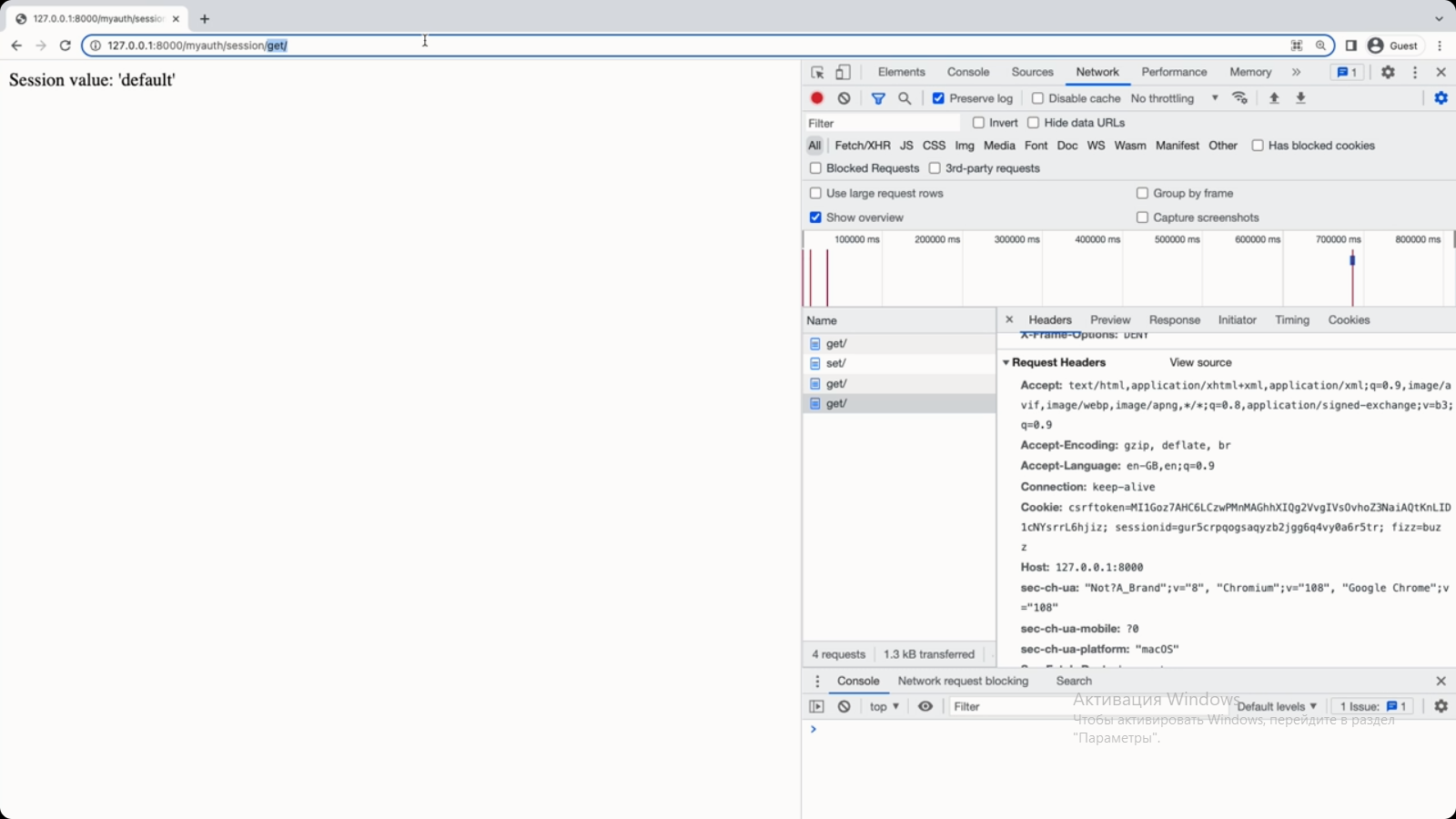
Создаем новую view функцию которая будет устанавливать куки. Для этого объявляем set\_cookie\_view

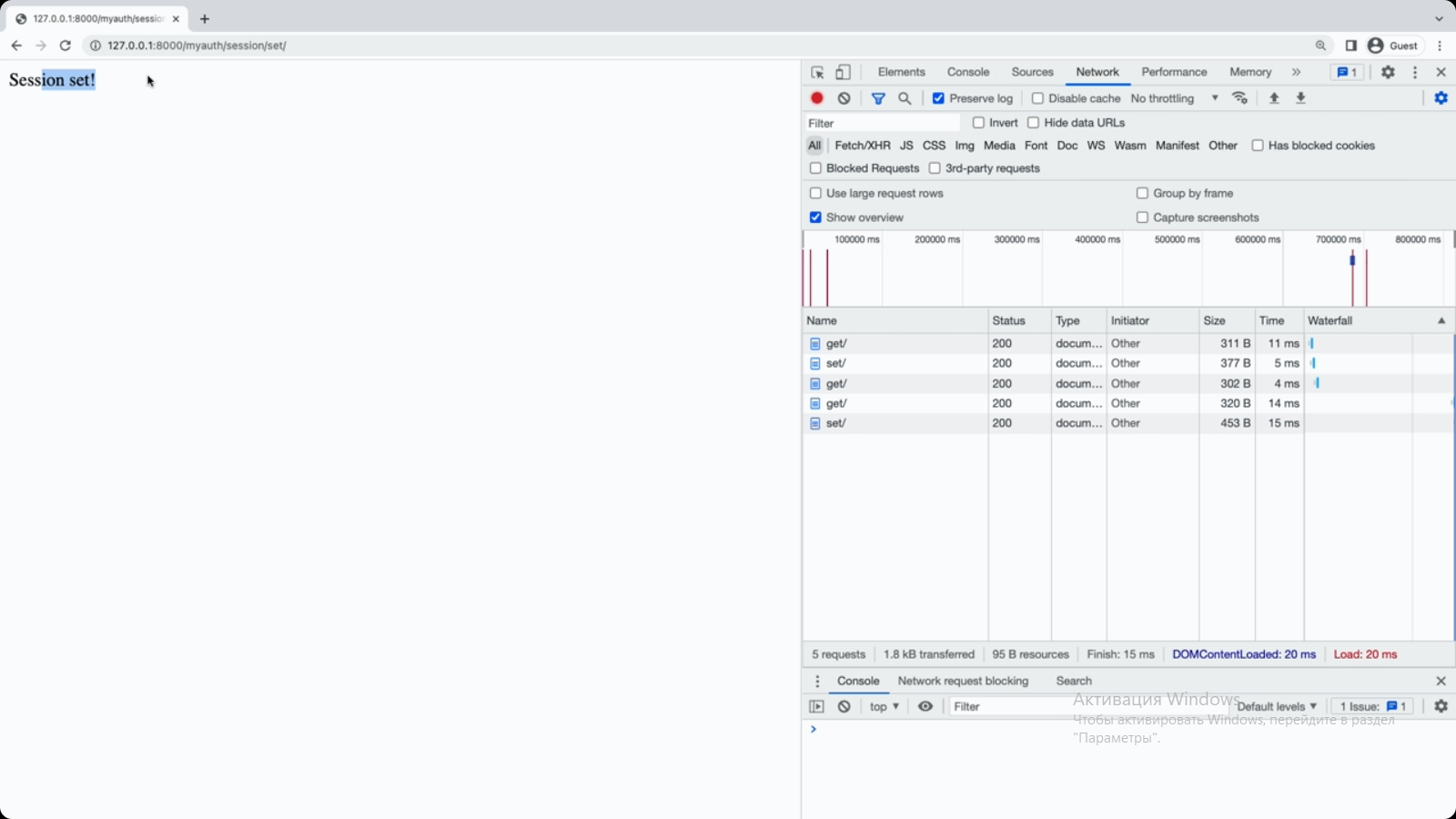
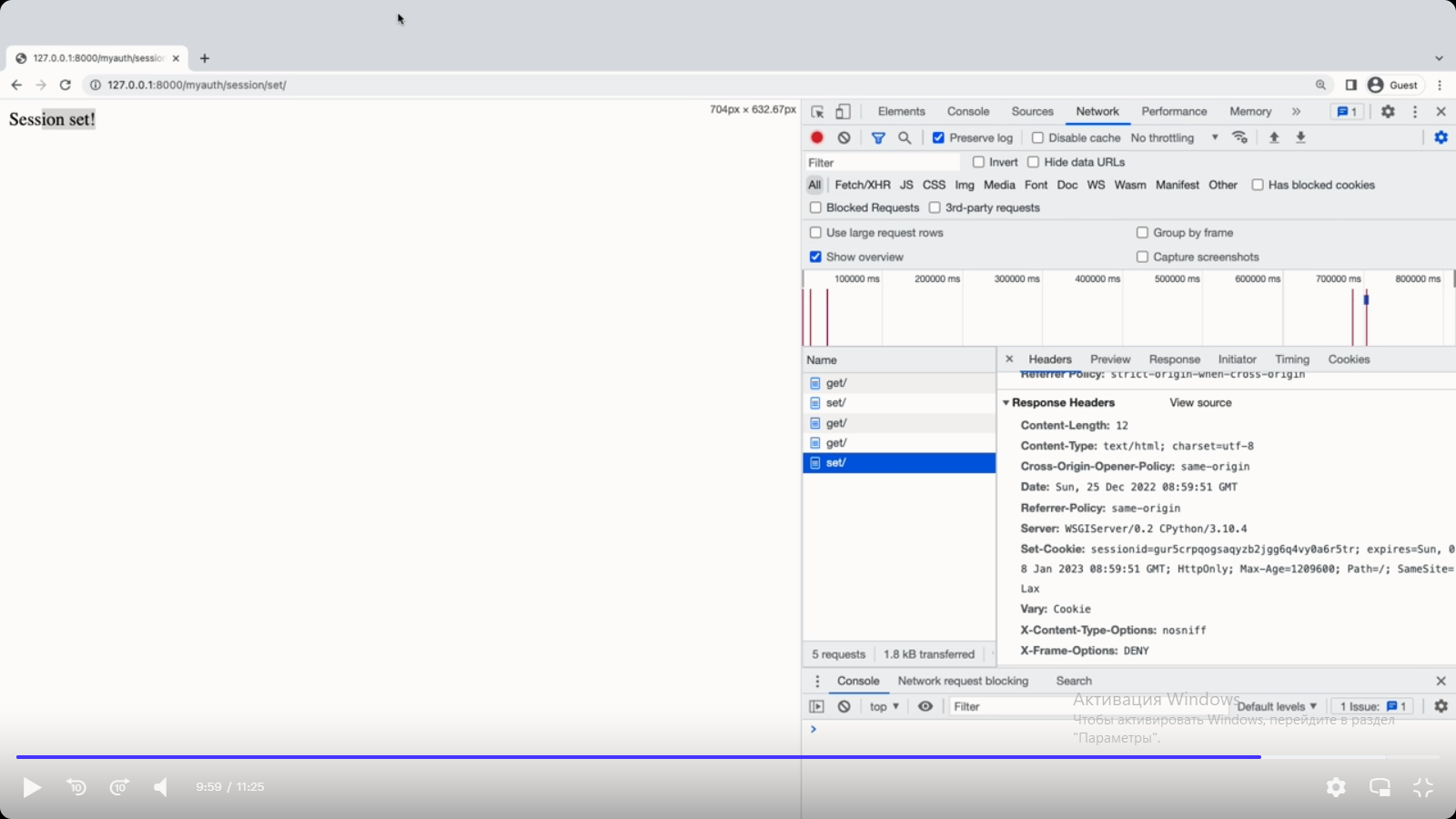
Создаём новый объектmax\_age=3600 это время жизни в секундах.

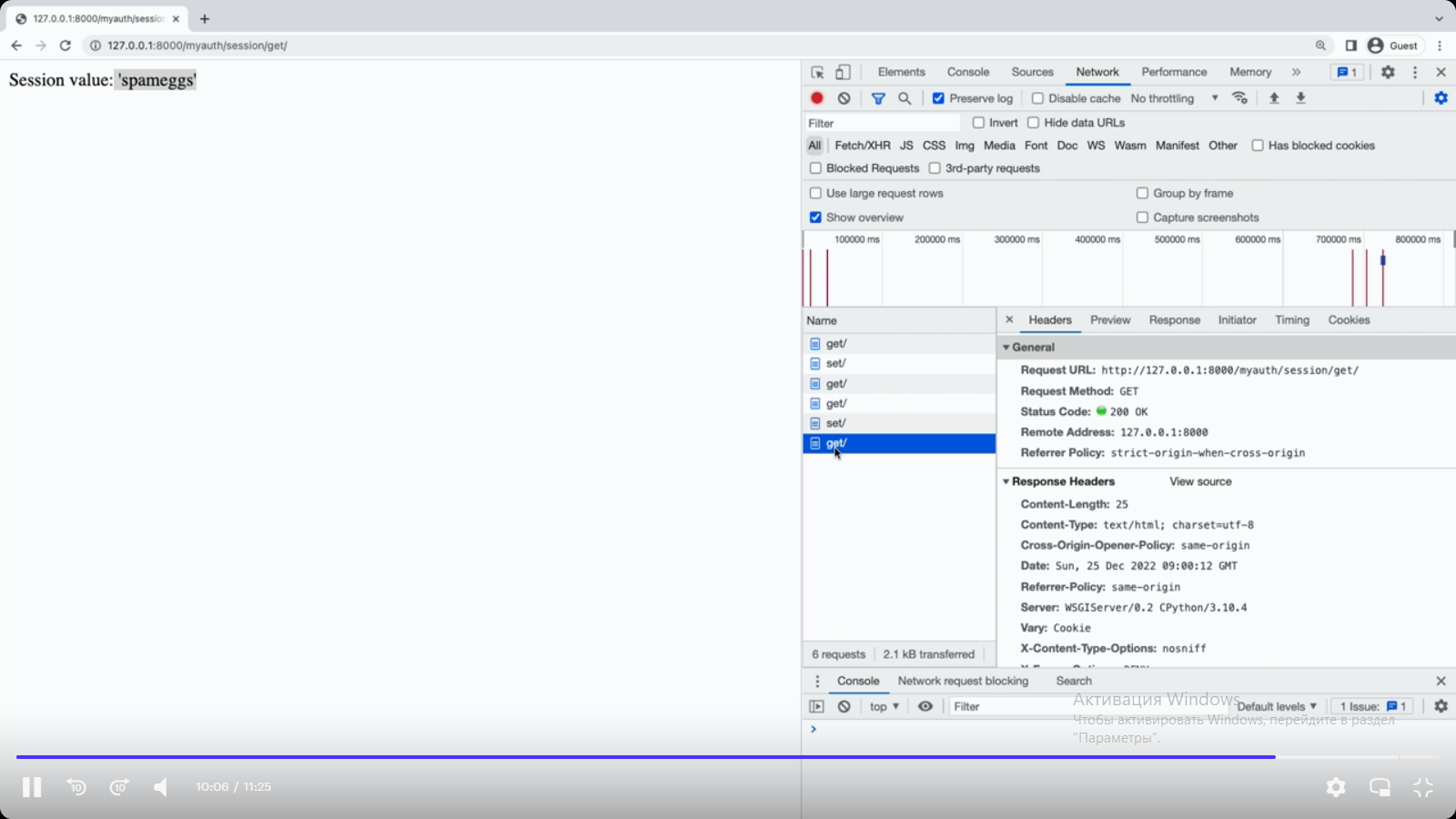
Добавляем view для чтения кукиСохраняем изменения и подключаем эти изменения в urls. Для этого импортируем их из views и указываем путьСохраняем изменения и переходим в браузер’default value’ потому, что ни какого значения не лежитв панели разработчика мы видим fizz=buzz и время когда оно сгорает. Мы поставили обратный отсчёт и эти данные пришли в браузер. Теперь если перейти опять на /get, то получим ‘buzz’в панели видим токен для безопасности, также id сессии и fizz=buzz.

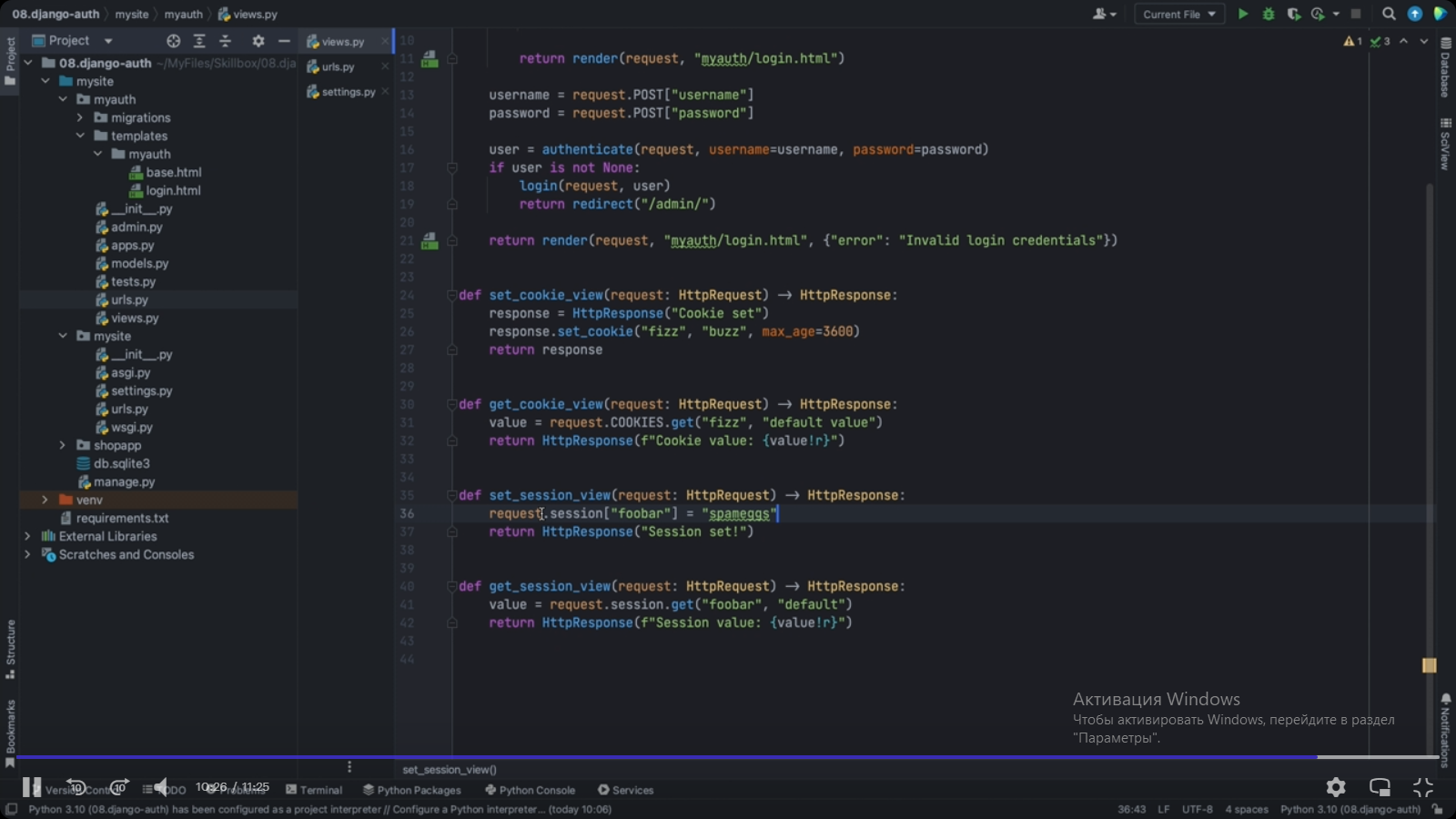
Сессии или сеансы - это способ хранения информации о текущей сесси или сеансе пользователя. Т.е. вы можете сохранить информацию о конкретном пользователе и использовать её, когда пользователь обращается к приложению. Например, вы можете использовать сессию для хранения информации о том какие страницы посещал пользователь или корзину, которую он наполняет на вашем сайте. Сами данные хранятся на бэкенде, т.е. в Django приложении, а у пользователя хранится только токен сессии и по этому токену, при каждом запросе, приложение достает необходимые данные. Токен хранится в браузере, т.е. в куке и он связывает пользователя с его сессией в приложении. Токен отправляется каждый раз при запросе к приложению и т.к. он уникален, то он используется ещё и для идентификации пользователя.

Django поддерживает несколько способов хранения сессий: хранение в БД, хранение в виде файлов или просто кэше. Вы можете указать необходимый cпособ хранения в настройках приложения Django. Чтобы использовать сессии в Django нужно убедиться, что midleweare подключён в настройках djangoТеперь можно использовать сессии в запросах, для этого создаём новую view функцию

Добавляем view для чтения сессииСохраняем и подключаем к urlsВ панели видим sessionid 

Если перейдем на set, то увидим что сессия установлена 

Однако если перейдём на get, то получим значение и это значение хранится не в браузере а на бэкенде. А в браузере только по id сессии проиходит запрос данных пользователя.

Обратите внимание на то, что данные в сессии сохраняются автоматически, т.е. здесь не нужно было делать ни какого save или в оличие от set\_cookie не нужно делать явную установку.  Т.е. вы сохраняете данные в словарь и django занимается тем, чтобы сохранить эти данные у себя. Также вам не нужно устанавливать время жизни токена сессии т.к. это делается автоматически, однако вы можете установить время жизни в настройках своего проекта.

Django предоставляет удобный набор инструментов для управления сессиями, т.е. сеансами пользователя при помщи сохранения токена сессии в куках. Сессии позволяют сохранять данные пользователя между запросами при этом не требуя сохранения всей информации в браузере пользователя. Данные будут храниться на бэкенде в django приложении, при этом не важно как (в кэше, в БД или в файлах), а у пользователя будет только токен сессии. Это позволяет сохранить больше информации о пользователе нежелb могло влесть в куки, а также это более безопасно, т.к. на стороне пользователя храниться один токен и django постоянно этот токен обновляет ратируя его по сображениям безопасности.

Описание

[How to use sessions | Django documentation](https://docs.djangoproject.com/en/4.1/topics/http/sessions/)