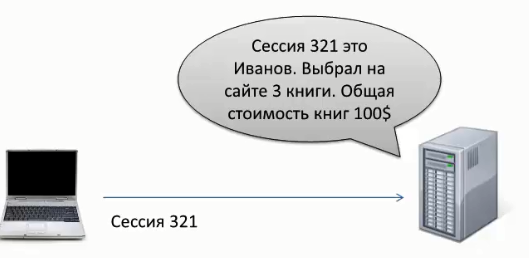
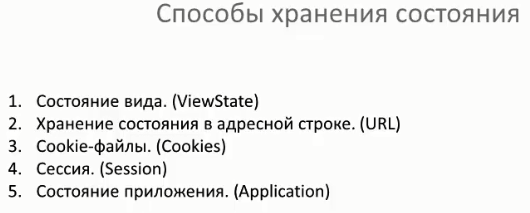


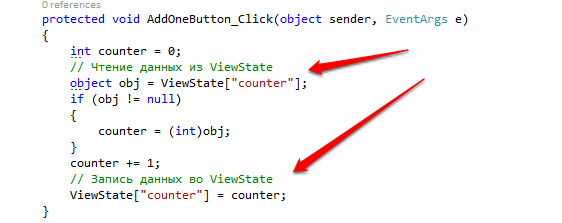
Cookie фалы



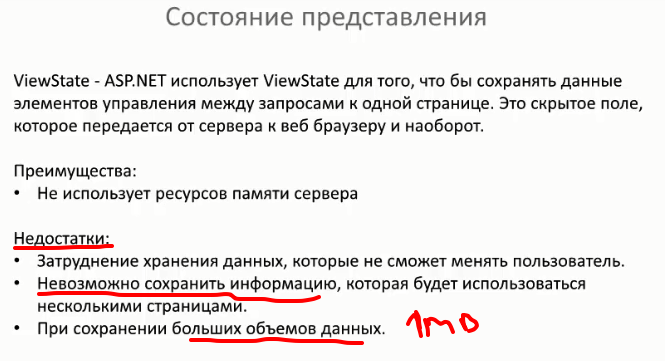


Использование ViewState для хранения состояния

ViewState["counter"];

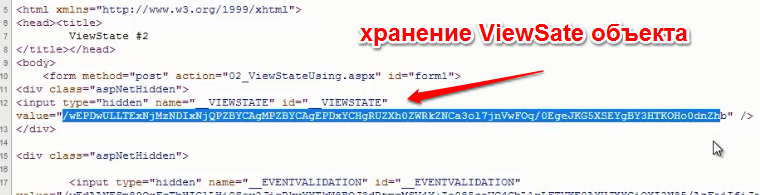


Живет коллекция пока есть postback запрос

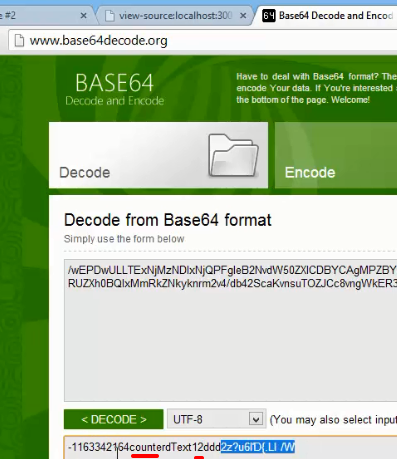


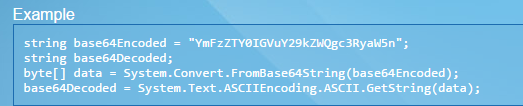




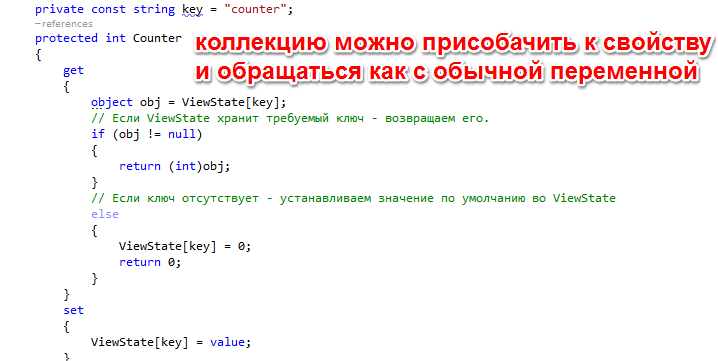


Декодировать

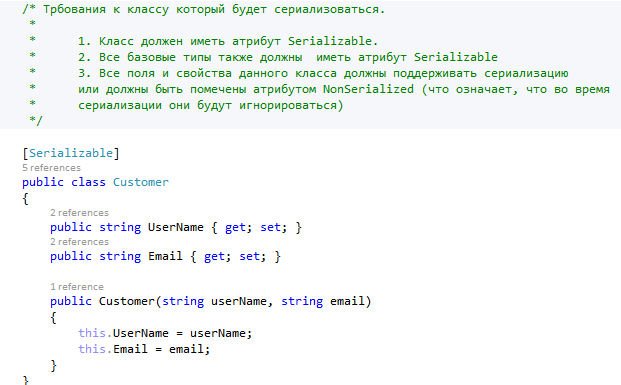


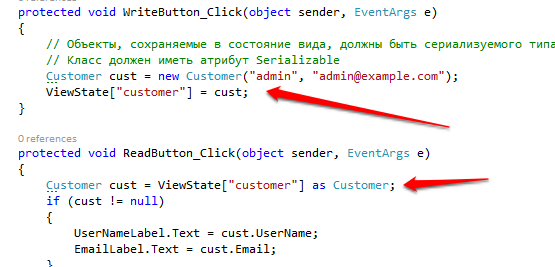


Оптимизированное решение



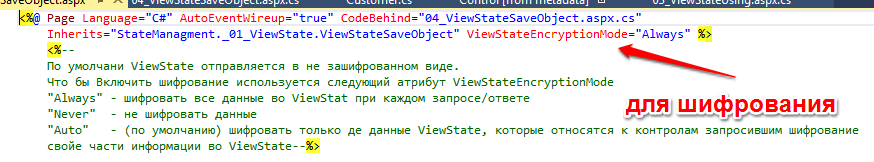
Либо пользовательские данные передавать





ViewStateEncryptionMode="Always"

Кодирование данных

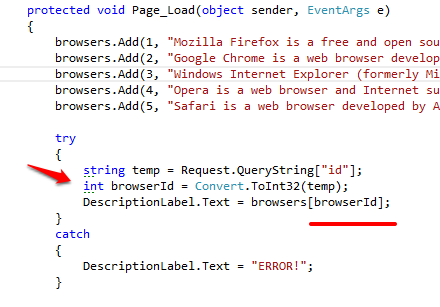


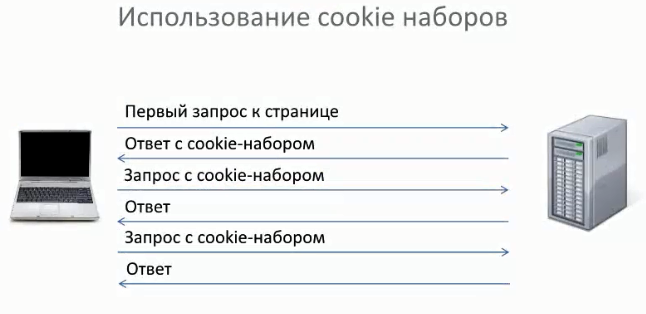
Второй вариант сохранения данных использование адресной строки, Requst.QueryString

Get запросы



На второй форме по ключу получаем значение





HttpCookie



protected void SaveButton\_Click(object sender, EventArgs e)

{

// Создаем cookie набор.

// Такой cookie-набор будет храниться до тех пор, пока пользователь не закроет браузер.

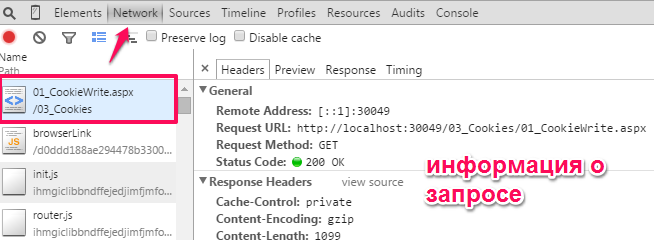
HttpCookie cookie = new HttpCookie("MyCookie", TextBox1.Text);

// Добавляем cookie-набор в текущий ответ.

Response.Cookies.Add(cookie);

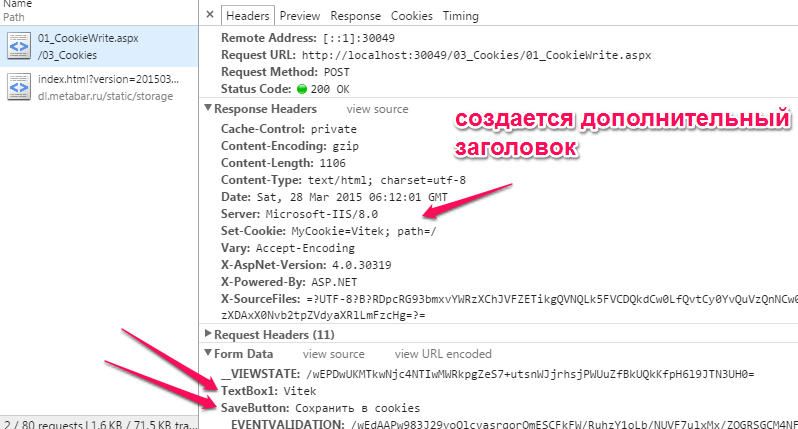
}

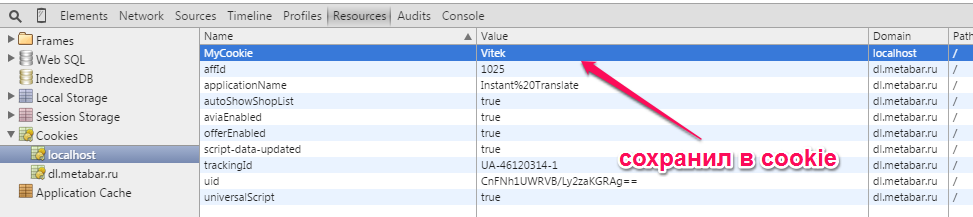
Информация по обычной загрузке



После нажатия на кнопку



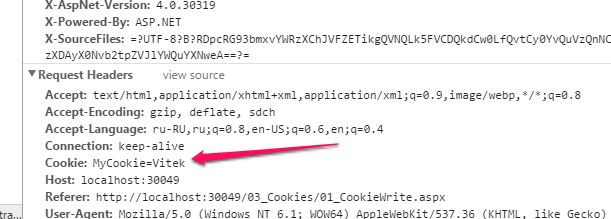




Ко всем страницам localhost, работает в контексте сессии (если браузер закроется то все данные пропадут), размер



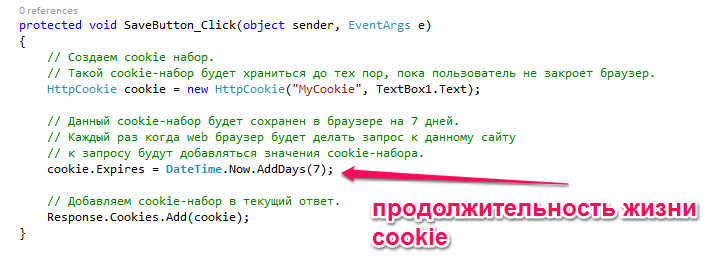
Перешли на другую страницу (ТЕСТИРОВАНИЕ)



Где с запроса от клиента получили cookie



cookie.Expires

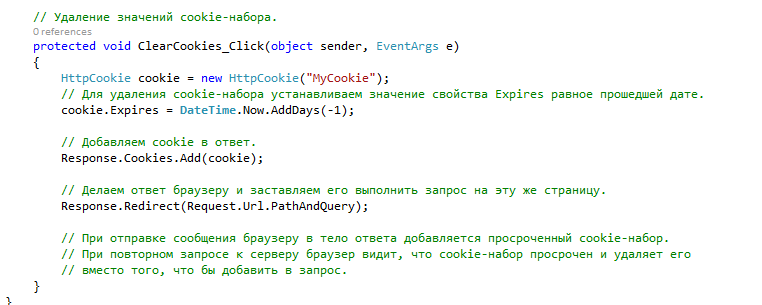




Удаление значений cookie

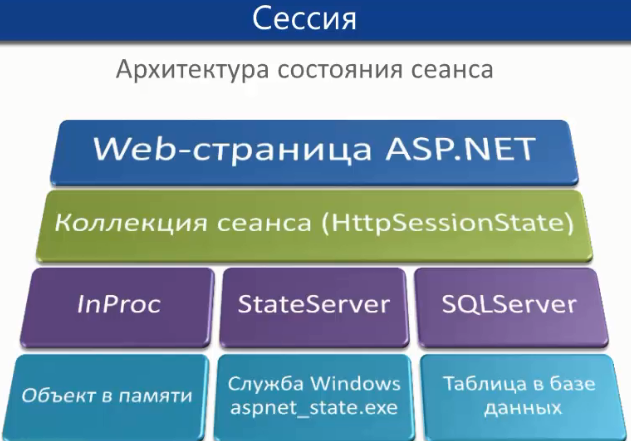
Request.Url.PathAndQuery – путь к странице

Response.Redirect - перейти



Cookie можно создать не более 4 кб

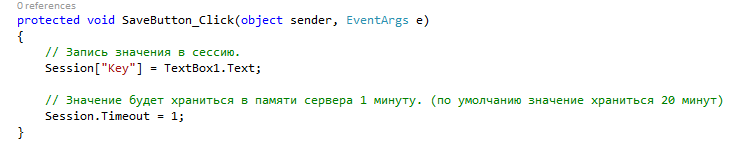
Большие размеры не приветствуются т.к. они передаются в заголовках от клиента, тратится время на передачу



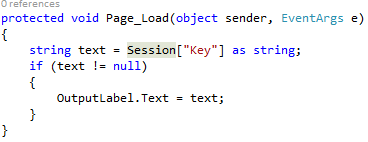
inProc – хранится в оперативной памяти на стороне компьютера сервера

stateserver – хранится в оперативной памяти отдельного компьютера (другого сервера, не того которое использует наше приложение)

Session["Key"]

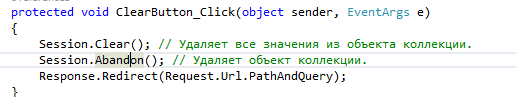


Считывание данных



Session.Clear(); // Удаляет все значения из объекта коллекции. (очищает коллекцию)

Session.Abandon(); // Удаляет объект коллекции. (саму коллекцию)



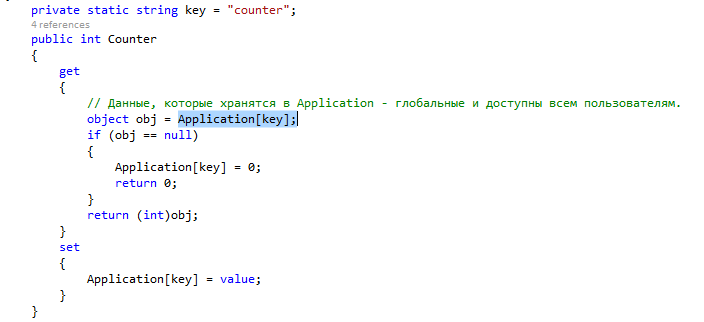


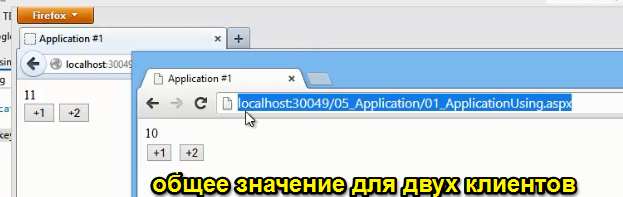
Сессия сохранила уникальный идентификатор клиента в cookie



Application[key];

Данные, которые хранятся в Application - глобальные и доступны всем пользователям.

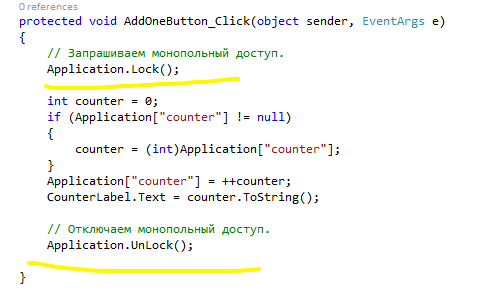




Разделяемый ресурс как и в потоках

Application.Lock(); - Запрашиваем монопольный доступ

Application.UnLock();

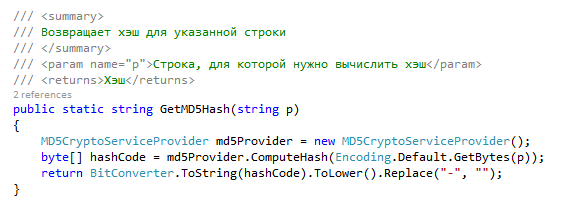




Пример авторизации на сайте

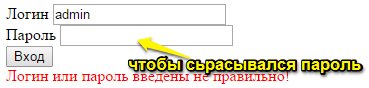
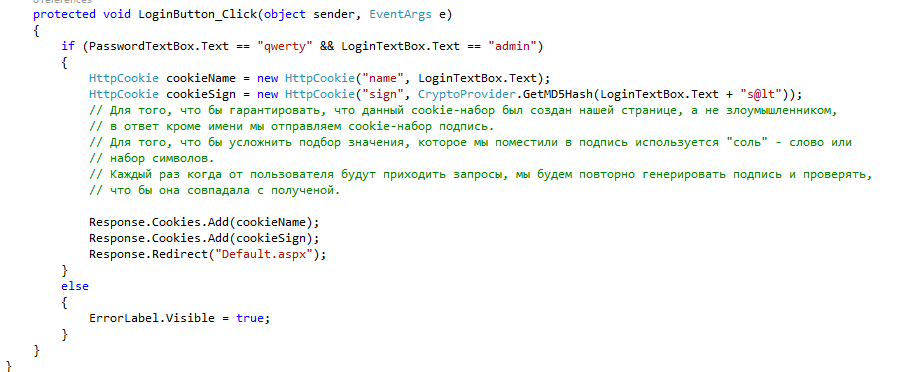
Алгоритм MD5 только для получения значения хэш обратно преобразовать не получиться

Каждый полученный байт будет разделен чертой и мы ее удаляем

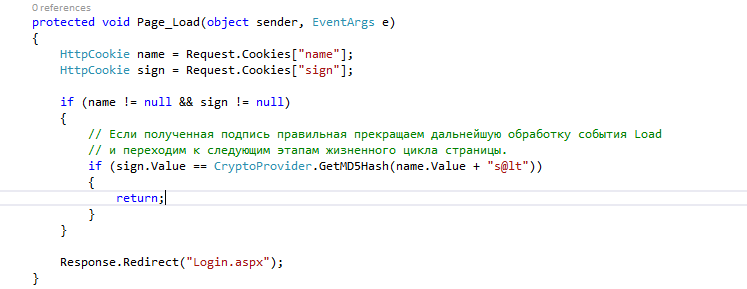


На странице логин



При попытки попасть на секретную страницу, если подпись не верная, то отправляем на страницу с регистрацией



Страница входа

