



## Урок Спринт 2.12

# Кеширование сессий

Управление сессиями — важная часть многих веб-приложений. Сессия хранит информацию о пользовательском взаимодействии с приложением. Кеширование сессий может значительно улучшить производительность, уменьшая нагрузку на сервера и базы данных.

Допустим, у нас есть приложение на Go, где мы хотим кешировать данные сессии. Мы можем использовать `middleware`, который будет проверять кеш перед выполнением запроса и сохранять данные сессии в кеш для быстрого доступа.

```
func SessionCacheMiddleware(next http.Handler) http.Handler {
    return http.HandlerFunc(func(w http.ResponseWriter, r *http.Request) {
        sessionId := getSessionIDFromRequest(r)
        if sessionData, found := cache.Get(sessionId); found {
            // Используем данные сессии из кеша
            processWithCachedSessionData(sessionData)
        } else {
            // Обрабатываем запрос без кешированной сессии
            next.ServeHTTP(w, r)
        }
    })
}
```

Этот код проверяет, есть ли данные сессии в кеше, и если они там, использует их, в противном случае продолжает обычную обработку запроса, создает сессию и записывает её в хранилище.

## Справка

Исключительное право на учебную программу и все сопутствующие ей учебные материалы, доступные в рамках сервиса, принадлежат АНО ДПО «Образовательные технологии Яндекса». Воспроизведение, копирование, распространение и иное использование программы и материалов допустимо только с предварительного письменного согласия АНО ДПО «Образовательные технологии Яндекса».

[Пользовательское соглашение.](#)

© 2018 – 2024 ООО «Яндекс»