

Текстовый вариант видеоурока из предыдущего шага

Следующим инструментом для синхронизации является таймер. Таймер позволяет нам настраивать тайм-ауты и ограничивать наши функции по времени выполнения. Создается он путем вызова функции `NewTimer` из пакета `Time`, в данном случае мы в переменную `Timer` присваиваем значение функции `NewTimer`, которая вернет ссылку на структуру таймер с ограничением в 3 секунды. Давайте на примерах кода посмотрим, как с этим работать. В нашей функции `main` мы как раз создаем эту структуру `Timer`. И, как мы можем видеть, функция `NewTimer` возвращает ссылку на структуру `Timer`. В данном случае нам уже не нужно брать ссылку, как мы это делали с `WaitGroup`, потому что здесь уже функция сама отдает ссылку.

У нас есть не буферизированный канал `quit`, который служит сигналом выхода. И мы в горутине запускаем функцию `jobWithTimeOut`. Давайте посмотрим, что происходит внутри. В этой функции мы используем инструкцию `select`. Смотрите, мы передали сюда наш `Timer`, и в кейсе слушаем канал из таймера. У таймера есть канал `C`, в который, после окончания тайм-аута, который мы настроили вот здесь, приходит сигнал. То есть после того, как 3 секунды завершатся, мы в этот канал получим сообщение. Собственно, это будет сигнализировать нам о том, что таймер закончился и время вышло. Поэтому мы и выводим на экран «ваше время вышло». При этом, если мы получаем сигнал для выхода, мы останавливаем наш таймер и печатаем, что таймер остановлен.

Давайте посмотрим на результат этой программы. Смотрите, сейчас мы попали в `default`, завершение функции, почему так произошло? Мы настроили тайм-аут на 3 секунды, а функция наша, как мы видим, выполнилась за 1 секунду, соответственно, тайм-аут у нас не сработал, потому что мы уложились в эти 3 секунды. Давайте поменяем наш тайм-аут на 1 секунду и посмотрим, что

изменится. Здесь уже мы получаем сообщение, что время вышло, потому что горютина `jobWithTimeOut` не справилась за 1 секунду, ну, потому что 1 секунду она просто спала.

Таким образом, с помощью инструмента таймер очень удобно настраивать тайм-ауты для наших методов, чтобы, например, наши клиенты не ждали подолгу выполнение какой-то функции, мы настраиваем там тайм-аут и отдаем сообщение об ошибке, если мы вышли за пределы каких-то границ времени ожидания.