Функции ***промежуточной обработки*** (middleware) - это функции, имеющие доступ к [объекту запроса](https://expressjs.com/ru/4x/api.html#req) (req), [объекту ответа](https://expressjs.com/ru/4x/api.html#res) (res) и к следующей функции промежуточной обработки в цикле “запрос-ответ” приложения. Следующая функция промежуточной обработки, как правило, обозначается переменной next.

Функции промежуточной обработки могут выполнять следующие задачи:

* Выполнение любого кода.
* Внесение изменений в объекты запросов и ответов.
* Завершение цикла “запрос-ответ”.
* Вызов следующей функции промежуточной обработки из стека.

Если текущая функция промежуточной обработки не завершает цикл “запрос-ответ”, она должна вызвать next() для передачи управления следующей функции промежуточной обработки. В противном случае запрос зависнет.

Приложение Express может использовать следующие типы промежуточных обработчиков:

* [Промежуточный обработчик уровня приложения](https://expressjs.com/ru/guide/using-middleware.html#middleware.application)
* [Промежуточный обработчик уровня маршрутизатора](https://expressjs.com/ru/guide/using-middleware.html#middleware.router)
* [Промежуточный обработчик для обработки ошибок](https://expressjs.com/ru/guide/using-middleware.html#middleware.error-handling)
* [Встроенные промежуточные обработчики](https://expressjs.com/ru/guide/using-middleware.html#middleware.built-in)
* [Промежуточные обработчики сторонних поставщиков ПО](https://expressjs.com/ru/guide/using-middleware.html#middleware.third-party)

https://expressjs.com/ru/guide/using-middleware.html

[https://metanit.com/web/nodejs/4.2.php](https://vk.com/away.php?utf=1&to=https%3A%2F%2Fmetanit.com%2Fweb%2Fnodejs%2F4.2.php)