1. Асинхронность в программировании.

В JS код выполняется последовательно сверху вниз (синхронно). Но чем сложнее код, тем становиться важно, чтобы интерфейс оставался быстрым и отзывчивым для пользователя. Если бы не была придуманна асинхронность, пользователям (да и программистам) приходилось бы долго ждать, пока не выполнится первый код, затем второй, затем третий, десятый……сотый код. При внедрении асинхронности в JS, многие задачи можно выполнять параллельно, то есть не дожидаясь, пока выполнится тот или иной код. При этом можно выполнять долгие сетевые запросы без блокирования основного потока. Таким образом, использование асинхронных функций оптимизирует код.

2. EventLoop.

При внедрении механизма асинхронности программисты столкнулись с такой проблемой как беспорядочность. Ведь при использовании асинхронности необходимо было применять некие правила. Для этого был внедрен механизм EventLoop, который контролирует очереди обратных вызовов, то есть EventLoop занимается организационными моментами. Другими словами, механизм EventLoop помогает JS обрабатывать асинхронный код.