**LazyLoad**

**Выполнение кода**

Выполнение кода по стандарту происходит по очереди строк сверху вниз.

Но это редко когда походит для сложных программ, где, например, изображение рендерится в фоновом режиме.

**Асинхронность**

Определение: когда код выполняется в одном потоке, но не по порядку.

Timer - частный случай асинхронности.

Асинхронный код – это код, работающий в том потоке, котором его вызвали. Это может быть главный или другой поток. Асинхронный код выполняется, когда его вызвали.

**Поток (Thread) в нашем приложении это реальный поток на нашем устройстве.**

Поток всегда выполняет одну операцию.

Время на создание потока может превысить время выполнения операции.

Кол-во реальных потоков равняется кол-ву ядер. Если создать 10 потоков на 2ух ядерной машине, все сведется к ожиданию потоков друг друга.

Потоки одна из первых технологий, которая появилась в ios для работы с многопоточностью.

**Многопоточность**

В отличии от асинхронности в многопоточной среде код выполняется не потом, по наступлению определенного события, а параллельно другим операциям основного потока в своем потоке.

На создание параллельного потока уходит определенное время.

**Grand Central Dispatch (****GCD центральный диспетчер задач)**

В случае с **GCD** мы лишь говорим, что этот блок кода должен выполняться параллельно с таким - то приоритетом, а уже **GCD** сам решает сколько потоков создать и как их разместить.

Параллельная очередь (concurrent queue)

NSOpertion

Представляет собой самый верхнеуровневый способ управления задачами.

**Приоритет**

Приоритет показывает сколько времени уделит процессор потоку. Чем больше приоритет, тем больше времени.

QoS – quality of service класс обслуживания