Чтобы переопределить метод в производном классе, этот метод должен быть снабжен модификатором **override**. Если этого не сделать, то компилятор предупредит о необходимости указания в объявлении производного метода либо модификатора new, либо модификатора override. По умолчанию компилятор подразумевает использование модификатора new, что,

вероятно, даст эффект, противоположный тому, что можно было ожидать

**Процесс** — это исполняемая программа, которой выделено собственное ограниченное адресное пространство.

**Синхронизация** — это организация процесса последовательного выполнения операций

**Lock**-Блокирует вызывающий поток до тех пор, пока этот поток не получит права владельца объекта mutex.

**Поток** — это "нить" процесса, которой отводится не собственное адресное пространство (память, проще говоря, оперативная), а часть пространства того процесса, что этот поток породил

**Мью́текс**— аналог одноместного семафора, служащий в программировании для синхронизации одновременно выполняющихся потоков.

**object\_thread».join(**) - запуск в «синхронном» режиме.

**object\_thread.detach()** - запуск в «асинхронном» режиме.

**Синхронно** — выполнение операций последовательно.

**Асинхронно** — «одновременно»

- объект ядра ОС, по сути позволяет блокировать свое состояние одному потоку и возвращать свой статус другим потокам.