**30. Ветвление в системах контроля версий.**

Почти каждая **система контроля версий (СКВ)** в какой-то форме поддерживает ветвление. Используя ветвление, Вы отклоняетесь от основной линии разработки и продолжаете работу независимо от нее, не вмешиваясь в основную линию. Во многих **СКВ создание веток** — это очень затратный процесс, часто требующий создания новой копии директории, что может занять много времени для большого проекта. **Ветвление Git** быстрее: операция создания ветки выполняется почти мгновенно, переключение между ветками туда-сюда, обычно, также быстро. В отличие от многих других **СКВ**, **Git** поощряет процесс работы, при котором ветвление и слияние выполняется часто

**Git** не хранит данные в виде последовательности изменений, он использует набор снимков (snapshot). Когда вы делаете коммит, Git сохраняет его в виде объекта, который содержит указатель на снимок (snapshot) подготовленных данных. Этот объект так же содержит имя автора и email, сообщение и указатель на коммит.

Во время индексации вычисляется **контрольная сумма** каждого файла, затем каждый файл сохраняется в репозиторий, а контрольная сумма попадает в **индекс**. Когда вы создаете коммит командой git commit, **Git** вычисляет **контрольные суммы** каждого подкаталога и сохраняет его в р**епозитории** как объект дерева каталогов.  Затем Git создает объект коммита с метаданными и указателем на основное дерево.

Ваш репозиторий Git теперь хранит пять объектов: **три блоб объекта** (по одному на каждый файл), **объект дерева каталогов** (содержащий список файлов и соответствующих им блобов) а так же **объект коммита**, содержащий метаданные и указатель на объект дерева каталогов.

Если вы сделаете изменения и создадите еще один коммит, тогда он будет содержать указатель на предыдущий коммит. **Ветка в Git** — это простой перемещаемый указатель на один из таких коммитов. По умолчанию, имя основной ветки в Git — **master**. Как только вы начнёте создавать коммиты, ветка master будет всегда указывать на **последний коммит.**

При **создании новой ветки (git branch)** создается **новый указатель** для дальнейшего перемещения. В результате создается новый указатель на текущий коммит.

То, в какой ветке находится пользователь GIT определяет по указателю **HEAD**. в **Git** концепция HEAD значительно **отличается** от других **систем контроля версий**, которые вы могли использовать **раньше (Subversion или CVS).** В Git — это указатель на текущую локальную ветку. В нашем случае мы все еще находимся в ветке master. Команда **git branch**только создает новую ветку, но не переключает на неё.

Для **переключения** на другую существующую ветку выполняется команда **git checkout**, она перемещает указатель HEAD.

То есть, это работает так, что вы создали ветку и переключились на нее, поработали, а затем вернулись в основную ветку и поработали в ней. Эти изменения изолированы друг от друга: вы можете свободно переключаться туда и обратно, а когда понадобится — объединить их.

Создание новой ветки в Git происходит быстро, в других **системах контроля версий** все файлы проекта копируются в другой **подкаталог**( это занимает больше времени)