Resetting Head Commit

Suppose, you notice a small error in a recent commit or a set of commits and want to redo that part without showing the undo in the history. In this case resetting the latest commit in the current branch (*HEAD*) to an earlier commit is helpful. The *Reset HEAD* operation sets the current *HEAD* to the specified commit and optionally resets the index and working tree to match.

To reset the current head to an earlier commit

1. On the main menu, choose **VCS | Git | Reset HEAD**. The Reset Head dialog box opens.

2. From the **Git Root** drop-down list, choose the required local repository.

3. Make sure the **Current Branch** read-only field shows the branch in which you actually want to reset the *HEAD*. If necessary, switch to the correct branch.

4. From the **Reset Type** drop-down list, select the desired reset strategy to apply:

   ■ To have the changed files preserved but not marked for commit, select the **Mixed** option. The index will be reset while the working tree will not and you will get a report of what has not been updated.

   > This is the default strategy.

   ■ To have only the *HEAD* pointer moved without updating the index and the working tree, select the **Soft** option. Your current state with any changes will remain different from the commit you are switching to.

   ■ To have both the working directory and the index changed to the specified commit, select the **Hard** option.

5. In the **To Commit** text box, specify the commit you want to reset the current *HEAD* to. Type its commit hash or use an expression, for example, of the following structure:

   ```
   <branch>~<number of commits backwards between the latest commit (HEAD) and the required
   commit>.
   ```

   Refer to the Git commit naming 🔗 conventions for details.

6. To check that the specified commit exists and is the one you actually need, click the **Validate** button. The **Paths affected in commit** dialog box shows the files that were affected in the specified commit. View the information and click **OK**.

   > If you specify a commit that does not exist, IntelliJ IDEA displays an error message.

See Also

Concepts:

■ Version Control with IntelliJ IDEA

Procedures:

■ Committing Changes to a Local Git Repository

Reference:

■ Version Control Reference

■ Reset Head Dialog

Web Resources:

■ Developer Community 🔗