The following variables are available to shell scripts

BRANCH_NAME

> For a multibranch project, this will be set to the name of the branch being built, for example in case you wish to deploy to production from `master` but not from feature branches; if corresponding to some kind of change request, the name is generally arbitrary (refer to `CHANGE_ID` and `CHANGE_TARGET`).

CHANGE_ID

> For a multibranch project corresponding to some kind of change request, this will be set to the change ID, such as a pull request number, if supported; else unset.

CHANGE_URL

> For a multibranch project corresponding to some kind of change request, this will be set to the change URL, if supported; else unset.

CHANGE_TITLE

> For a multibranch project corresponding to some kind of change request, this will be set to the title of the change, if supported; else unset.

CHANGE_AUTHOR

> For a multibranch project corresponding to some kind of change request, this will be set to the username of the author of the proposed change, if supported; else unset.

CHANGE_AUTHOR_DISPLAY_NAME

> For a multibranch project corresponding to some kind of change request, this will be set to the human name of the author, if supported; else unset.

CHANGE_AUTHOR_EMAIL

> For a multibranch project corresponding to some kind of change request, this will be set to the email address of the author, if supported; else unset.

CHANGE_TARGET

> For a multibranch project corresponding to some kind of change request, this will be set to the target or base branch to which the change could be merged, if supported; else unset.

CHANGE_BRANCH

> For a multibranch project corresponding to some kind of change request, this will be set to the name of the actual head on the source control system which may or may not be different from `BRANCH_NAME`. For example in GitHub or Bitbucket this would have the name of the origin branch whereas `BRANCH_NAME` would be something like `PR-24`.

CHANGE_FORK

> For a multibranch project corresponding to some kind of change request, this will be set to the name of the forked repo if the change originates from one; else unset.

TAG_NAME

> For a multibranch project corresponding to some kind of tag, this will be set to the name of the tag being built, if supported; else unset.

TAG_TIMESTAMP

> For a multibranch project corresponding to some kind of tag, this will be set to a timestamp of the tag in milliseconds since Unix epoch, if supported; else unset.

TAG_UNIXTIME

> For a multibranch project corresponding to some kind of tag, this will be set to a timestamp of the tag in seconds since Unix epoch, if supported; else unset.

TAG_DATE

> For a multibranch project corresponding to some kind of tag, this will be set to a timestamp in the format as defined by java.util.Date#toString() (e.g., Wed Jan 1 00:00:00 UTC 2020), if supported;

else unset.

BUILD_NUMBER
The current build number, such as "153"

BUILD_ID
The current build ID, identical to BUILD_NUMBER for builds created in 1.597+, but a YYYY-MM-DD_hh-mm-ss timestamp for older builds

BUILD_DISPLAY_NAME
The display name of the current build, which is something like "#153" by default.

JOB_NAME
Name of the project of this build, such as "foo" or "foo/bar".

JOB_BASE_NAME
Short Name of the project of this build stripping off folder paths, such as "foo" for "bar/foo".

BUILD_TAG
String of "jenkins-${JOB_NAME}-${BUILD_NUMBER}". All forward slashes ("/") in the JOB_NAME are replaced with dashes ("-"). Convenient to put into a resource file, a jar file, etc for easier identification.

EXECUTOR_NUMBER
The unique number that identifies the current executor (among executors of the same machine) that's carrying out this build. This is the number you see in the "build executor status", except that the number starts from 0, not 1.

NODE_NAME
Name of the agent if the build is on an agent, or "master" if run on master

NODE_LABELS
Whitespace-separated list of labels that the node is assigned.

WORKSPACE
The absolute path of the directory assigned to the build as a workspace.

WORKSPACE_TMP
A temporary directory near the workspace that will not be browsable and will not interfere with SCM checkouts. May not initially exist, so be sure to create the directory as needed (e.g., `mkdir -p` on Linux). Not defined when the regular workspace is a drive root.

JENKINS_HOME
The absolute path of the directory assigned on the master node for Jenkins to store data.

JENKINS_URL
Full URL of Jenkins, like `http://server:port/jenkins/` (note: only available if *Jenkins URL* set in system configuration)

BUILD_URL
Full URL of this build, like `http://server:port/jenkins/job/foo/15/` (*Jenkins URL* must be set)

JOB_URL
Full URL of this job, like `http://server:port/jenkins/job/foo/` (*Jenkins URL* must be set)

GIT_COMMIT
The commit hash being checked out.

GIT_PREVIOUS_COMMIT
The hash of the commit last built on this branch, if any.

GIT_PREVIOUS_SUCCESSFUL_COMMIT
The hash of the commit last successfully built on this branch, if any.

GIT_BRANCH
The remote branch name, if any.

GIT_LOCAL_BRANCH
The local branch name being checked out, if applicable.

GIT_CHECKOUT_DIR
> The directory that the repository will be checked out to. This contains the value set in Checkout to a sub-directory, if used.

GIT_URL
> The remote URL. If there are multiple, will be `GIT_URL_1`, `GIT_URL_2`, etc.

GIT_COMMITTER_NAME
> The configured Git committer name, if any, that will be used for FUTURE commits from the current workspace. It is read from the **Global Config user.name Value** field of the Jenkins **Configure System** page.

GIT_AUTHOR_NAME
> The configured Git author name, if any, that will be used for FUTURE commits from the current workspace. It is read from the **Global Config user.name Value** field of the Jenkins **Configure System** page.

GIT_COMMITTER_EMAIL
> The configured Git committer email, if any, that will be used for FUTURE commits from the current workspace. It is read from the **Global Config user.email Value** field of the Jenkins **Configure System** page.

GIT_AUTHOR_EMAIL
> The configured Git author email, if any, that will be used for FUTURE commits from the current workspace. It is read from the **Global Config user.email Value** field of the Jenkins **Configure System** page.