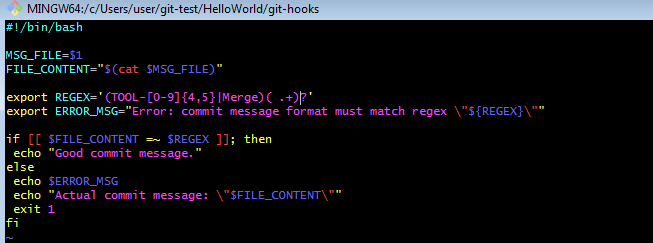
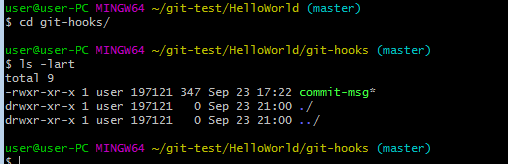
**How to make Pre-Commit Message hooks policy at Developer side**

STEP-1 : Clone the Repository

STEP- 2.

At repository root level, Create a directory for holding the scripts (let's say that the name will be: git-hooks)



**Code is below:**

#!/bin/bash

MSG\_FILE=$1

FILE\_CONTENT="$(cat $MSG\_FILE)"

export REGEX='(TOOL-[0-9]{4,5}|Merge)( .+)?'

export ERROR\_MSG="Error: commit message format must match regex \"${REGEX}\""

if [[ $FILE\_CONTENT =~ $REGEX ]]; then

echo "Good commit message."

else

echo $ERROR\_MSG

echo "Actual commit message: \"$FILE\_CONTENT\""

exit 1

fi

Add this file to master so that when developer create a feature branch he should have this folder and file in his feature branch.

**commands :**

git add .

git commit -m "TOOL-0000 Adjusting the hook"

git push origin master

git checkout feature-1

**Command:** This command should be executed from the root directory of your repository.

cd ../

git config core.hooksPath=./git-hooks

git config core.hooksPath ./git-hooks

Result:

This will help to apply the policy that no develope will allow to commit a message without **TOOL-\*\*\*\***  phrase in commit message.

This commands instructs (only for this single repository) git that the scripts acting as hooks are stored in the git-hooks directory.

From now on if you try to commit anything with the commit message not starting with TOOL-XXXX or TOOL-XXXXX (where X is a digit) the commit will fail with the proper message.