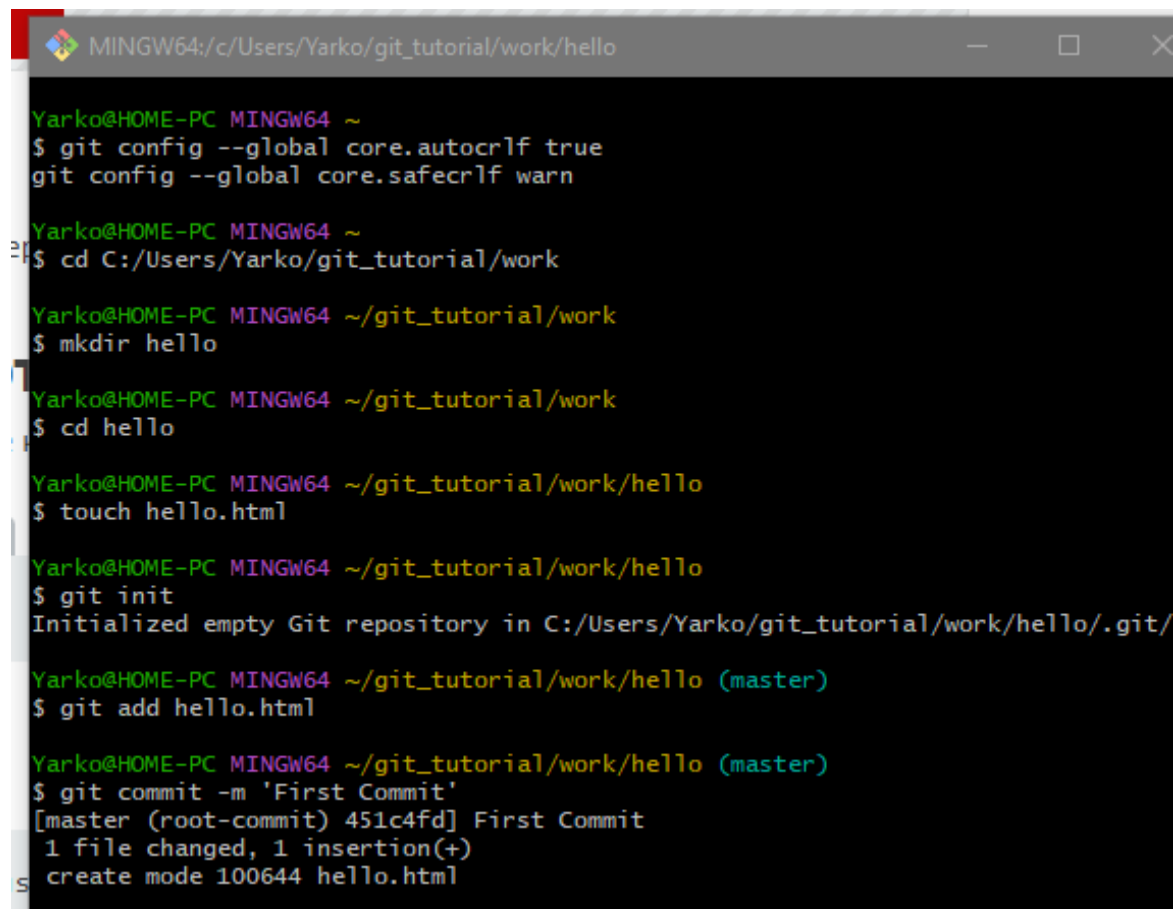


Attach the photo of your completed tasks.

A screenshot of a terminal window titled 'MINGW64:/c/Users/Yarko/git_tutorial/work/hello'. The terminal shows a series of commands and their outputs for setting up a Git repository. The user sets global Git configuration, navigates to the 'hello' directory, creates a new file 'hello.html', initializes a Git repository, adds the file, and makes the first commit.

```
MINGW64:/c/Users/Yarko/git_tutorial/work/hello

Yarko@HOME-PC MINGW64 ~
$ git config --global core.autocrlf true
git config --global core.safecrlf warn

Yarko@HOME-PC MINGW64 ~
$ cd C:/Users/Yarko/git_tutorial/work

Yarko@HOME-PC MINGW64 ~/git_tutorial/work
$ mkdir hello

Yarko@HOME-PC MINGW64 ~/git_tutorial/work
$ cd hello

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello
$ touch hello.html

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello
$ git init
Initialized empty Git repository in C:/Users/Yarko/git_tutorial/work/hello/.git/

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git add hello.html

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git commit -m 'First Commit'
[master (root-commit) 451c4fd] First Commit
1 file changed, 1 insertion(+)
create mode 100644 hello.html
```

```
Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git status
On branch master
nothing to commit, working tree clean

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   hello.html

no changes added to commit (use "git add" and/or "git commit -a")

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git add hello.html
warning: in the working copy of 'hello.html', LF will be replaced by CRLF the ne
xt time Git touches it

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   hello.html
```

```

marko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git commit
[master c0ea481] Added h1 tag
1 file changed, 12 insertions(+), 1 deletion(-)

marko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git status
On branch master
nothing to commit, working tree clean

marko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git add hello.html
warning: in the working copy of 'hello.html', LF will be replaced by CRLF the next time Git touches it

marko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   hello.html

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   hello.html

marko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git commit -m "Added standard HTML page tags"
[master 427fe0d] Added standard HTML page tags
1 file changed, 4 insertions(+), 11 deletions(-)

marko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   hello.html

no changes added to commit (use "git add" and/or "git commit -a")

marko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git add .
warning: in the working copy of 'hello.html', LF will be replaced by CRLF the next time Git touches it

marko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   hello.html

marko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git commit -m "Added HTML header"
[master 5b482a9] Added HTML header
1 file changed, 2 insertions(+)

```

```
Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git log
commit 5b482a92e2b14e4c1380ff69812ff70300b22b14 (HEAD -> master)
Author: Yaroslav Rudenko <betonbk@gmail.com>
Date: Tue Sep 27 16:51:36 2022 +0300

    Added HTML header

commit 427fe0d314711c85e9899792662b147e64347151
Author: Yaroslav Rudenko <betonbk@gmail.com>
Date: Tue Sep 27 16:49:44 2022 +0300

    Added standard HTML page tags

commit c0ea481d978bcadf9b4b4268e78b2e456c757e95
Author: Yaroslav Rudenko <betonbk@gmail.com>
Date: Tue Sep 27 16:41:25 2022 +0300

    Added h1 tag

commit 451c4fd4d643470bc1156b9be10e5505b27aa111
Author: Yaroslav Rudenko <betonbk@gmail.com>
Date: Tue Sep 27 16:28:11 2022 +0300

    First Commit

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git log --pretty=oneline
5b482a92e2b14e4c1380ff69812ff70300b22b14 (HEAD -> master) Added HTML header
427fe0d314711c85e9899792662b147e64347151 Added standard HTML page tags
c0ea481d978bcadf9b4b4268e78b2e456c757e95 Added h1 tag
451c4fd4d643470bc1156b9be10e5505b27aa111 First Commit
```

```
Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git log --pretty=format:"%h %ad | %s%d [%an]" --graph --date=short
* 5b482a9 2022-09-27 | Added HTML header (HEAD -> master) [Yaroslav Rudenko]
* 427fe0d 2022-09-27 | Added standard HTML page tags [Yaroslav Rudenko]
* c0ea481 2022-09-27 | Added h1 tag [Yaroslav Rudenko]
* 451c4fd 2022-09-27 | First Commit [Yaroslav Rudenko]

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git config --global alias.co checkout

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ ^[[200~git config --global alias.ci commit
bash: $'\E[200~git': command not found

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git config --global alias.ci commit

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git config --global alias.st status

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git config --global alias.br branch

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git config --global alias.hist "log --pretty=format:'%h %ad | %s%d [%an]' --graph --date=short"

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git config --global alias.type 'cat-file -t'

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git config --global alias.dump 'cat-file -p'

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ ..
```

```
Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git hist
* 5b482a9 2022-09-27 | Added HTML header (HEAD -> master) [Yaroslav Rudenko]
* 427Fe0d 2022-09-27 | Added standard HTML page tags [Yaroslav Rudenko]
* c0ea481 2022-09-27 | Added h1 tag [Yaroslav Rudenko]
* 451c4fd 2022-09-27 | First Commit [Yaroslav Rudenko]
```

```
Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git checkout 5b482a9
Note: switching to '5b482a9'.
```

You are in 'detached HEAD' state. You can look around, make experimental changes and commit them, and you can discard any commits you make in this state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may do so (now or later) by using `-c` with the switch command. Example:

```
git switch -c <new-branch-name>
```

Or undo this operation with:

```
git switch -
```

Turn off this advice by setting config variable `advice.detachedHead` to false

HEAD is now at 5b482a9 Added HTML header

```
Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello ((5b482a9...))
$ cat hello.html
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
```

```
Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello ((5b482a9...))
$ git checkout master
Switched to branch 'master'
```

```
Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ cat hello.html|
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
```

```
Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
```

```

$ git tag v1

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git checkout v1^
Note: switching to 'v1^'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 427fe0d Added standard HTML page tags

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello ((427fe0d...))
$ cat hello.html
<html>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello ((427fe0d...))
$ git tag v1-beta

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello ((v1-beta))
$ git checkout v1
Previous HEAD position was 427fe0d Added standard HTML page tags
HEAD is now at 5b482a9 Added HTML header

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello ((v1))
$ git checkout v1-beta
Previous HEAD position was 5b482a9 Added HTML header
HEAD is now at 427fe0d Added standard HTML page tags

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello ((v1-beta))
$ git tag
v1
v1-beta

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello ((v1-beta))
$ git hist master --all
* 5b482a9 2022-09-27 | Added HTML header (tag: v1, master) [Yaroslav Rudenko]
* 427fe0d 2022-09-27 | Added standard HTML page tags (HEAD, tag: v1-beta) [Yaroslav Rudenko]
* c0ea481 2022-09-27 | Added h1 tag [Yaroslav Rudenko]
* 451c4fd 2022-09-27 | First Commit [Yaroslav Rudenko]

```

```
Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello ((v1-beta))
$ git checkout master
Previous HEAD position was 427fe0d Added standard HTML page tags
Switched to branch 'master'

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   hello.html

no changes added to commit (use "git add" and/or "git commit -a")

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git checkout hello.html
Updated 1 path from the index

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git status
On branch master
nothing to commit, working tree clean

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ cat hello.html
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
```

```
Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git add hello.html

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   hello.html

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git reset HEAD hello.html
Unstaged changes after reset:
M       hello.html

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git checkout hello.html
Updated 1 path from the index

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git status
On branch master
nothing to commit, working tree clean
```



```

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git add hello.html

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git commit -m "Oops, we didn't want this commit"
[master a84e6ac] Oops, we didn't want this commit
1 file changed, 1 insertion(+)

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git revert HEAD
[master d978ca5] Revert "Oops, we didn't want this commit"
1 file changed, 1 deletion(-)

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git hist
* d978ca5 2022-09-27 | Revert "Oops, we didn't want this commit" (HEAD -> master) [Yaroslav Rudenko]
* a84e6ac 2022-09-27 | Oops, we didn't want this commit [Yaroslav Rudenko]
* 5b482a9 2022-09-27 | Added HTML header (tag: v1) [Yaroslav Rudenko]
* 427fe0d 2022-09-27 | Added standard HTML page tags (tag: v1-beta) [Yaroslav Rudenko]
* c0ea481 2022-09-27 | Added h1 tag [Yaroslav Rudenko]
* 451c4fd 2022-09-27 | First Commit [Yaroslav Rudenko]

```

```

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git hist
* d978ca5 2022-09-27 | Revert "Oops, we didn't want this commit" (HEAD -> master) [Yaroslav Rudenko]
* a84e6ac 2022-09-27 | Oops, we didn't want this commit [Yaroslav Rudenko]
* 5b482a9 2022-09-27 | Added HTML header (tag: v1) [Yaroslav Rudenko]
* 427fe0d 2022-09-27 | Added standard HTML page tags (tag: v1-beta) [Yaroslav Rudenko]
* c0ea481 2022-09-27 | Added h1 tag [Yaroslav Rudenko]
* 451c4fd 2022-09-27 | First Commit [Yaroslav Rudenko]

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git tag oops

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git reset --hard v1
HEAD is now at 5b482a9 Added HTML header

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git hist
* 5b482a9 2022-09-27 | Added HTML header (HEAD -> master, tag: v1) [Yaroslav Rudenko]
* 427fe0d 2022-09-27 | Added standard HTML page tags (tag: v1-beta) [Yaroslav Rudenko]
* c0ea481 2022-09-27 | Added h1 tag [Yaroslav Rudenko]
* 451c4fd 2022-09-27 | First Commit [Yaroslav Rudenko]

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$ git hist --all
* d978ca5 2022-09-27 | Revert "Oops, we didn't want this commit" (tag: oops) [Yaroslav Rudenko]
* a84e6ac 2022-09-27 | Oops, we didn't want this commit [Yaroslav Rudenko]
* 5b482a9 2022-09-27 | Added HTML header (HEAD -> master, tag: v1) [Yaroslav Rudenko]
* 427fe0d 2022-09-27 | Added standard HTML page tags (tag: v1-beta) [Yaroslav Rudenko]
* c0ea481 2022-09-27 | Added h1 tag [Yaroslav Rudenko]
* 451c4fd 2022-09-27 | First Commit [Yaroslav Rudenko]

Yarko@HOME-PC MINGW64 ~/git_tutorial/work/hello (master)
$

```