

GitLab: Setup sample Python repository

GitLab repository creation:

1. Go to project > New project > sample project name > Create Project

New project

A project is where you house your files (repositories), plan your work (issues), and publish your documentation (wikis), among other things.

All features are enabled for blank projects, from templates, or when importing, but you can disable them afterward in the project settings.

Tip: You can also create a project from the command line. [Show command](#)

Blank project | Create from template | Import project

Project name: Novartis-Test

Project URL: http://novartis.devopslatime root | Project slug: novartis-test

Want to house several dependent projects under the same namespace? [Create a group](#)

Project description (optional):

Visibility Level: Private (selected), Internal, Public

☐ Initialize repository with a README

Create project | Cancel

2. No Need to Create repository ,Automatically will create once Project created.
3. it will create with an empty repository.

To get started you will need to run below commands.

Configure Git for the first time

- 1.Setting your Git username for every repository on your computer.open git bash and run below commands

```
gbiswal@AIPL-LT-1025 MINGW64 /d/LuckyLuckyLucky/Altimetrik/DevOps-Platform-Implementation-and-Architecture/Code/git/altimetrik.com/bitbucket/projects/DEVOPS
$ git config --global user.name "gbiswal"

gbiswal@AIPL-LT-1025 MINGW64 /d/LuckyLuckyLucky/Altimetrik/DevOps-Platform-Implementation-and-Architecture/Code/git/altimetrik.com/bitbucket/projects/DEVOPS
$ git config --global user.email "gbiswal@altimetrik.com"
```

To check whether username is added or not run git config --list command

```
$ git config --list
core.symlinks=false
core.autocrlf=true
core.fscache=true
color.diff=auto
color.status=auto
color.branch=auto
color.interactive=true
help.format=html
rebase.autosquash=true
http.sslcainfo=C:/Users/gbiswal/AppData/Local/Programs/Git/mingw64/ssl/certs/ca-bundle.crt
http.sslbackend=openssl
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge --skip -- %f
filter.lfs.process=git-lfs filter-process --skip
filter.lfs.required=true
credential.helper=manager
user.name=gbiswal
user.email=gbiswal@altimetrik.com
```

Clone repository

1. If you want to simply clone this empty repository then run this command in your terminal.

```
gbiswal@AIPL-LT-1025 MINGW64 /d/LuckyLuckyLucky/Altimetrik/DevOps-Platform-Implementation-and-Architecture/Code/git/altimetrik.com/bitbucket/projects/DEVOPS
$ git clone http://novartis.devops.altimetrik.io:7080/root/novartis-test.git
Cloning into 'novartis-test'...
warning: You appear to have cloned an empty repository.
```

2. If you already have code ready to be pushed to this repository then run this in your terminal

```
$ cd novartis-test
$ touch README.md
$ git add README.md
$ git commit -m "add README"
$ git push -u origin master
```

```
gbiswal@AIPL-LT-1025 MINGW64 /d/LuckyLuckyLucky/Altimetrik/DevOps-Platform-Implementation-and-Architecture/Code/git/altimetrik.com/bitbucket/projects/DEVOPS
$ git clone http://novartis.devops.altimetrik.io:7080/root/novartis-test.git
Cloning into 'novartis-test'...
warning: You appear to have cloned an empty repository.
gbiswal@AIPL-LT-1025 MINGW64 /d/LuckyLuckyLucky/Altimetrik/DevOps-Platform-Implementation-and-Architecture/Code/git/altimetrik.com/bitbucket/projects/DEVOPS
$ cd novartis-test
gbiswal@AIPL-LT-1025 MINGW64 /d/LuckyLuckyLucky/Altimetrik/DevOps-Platform-Implementation-and-Architecture/Code/git/altimetrik.com/bitbucket/projects/DEVOPS/novartis-test
$ pwd
/d/LuckyLuckyLucky/Altimetrik/DevOps-Platform-Implementation-and-Architecture/Code/git/altimetrik.com/bitbucket/projects/DEVOPS/novartis-test
gbiswal@AIPL-LT-1025 MINGW64 /d/LuckyLuckyLucky/Altimetrik/DevOps-Platform-Implementation-and-Architecture/Code/git/altimetrik.com/bitbucket/projects/DEVOPS/novartis-test
$ touch README.md
gbiswal@AIPL-LT-1025 MINGW64 /d/LuckyLuckyLucky/Altimetrik/DevOps-Platform-Implementation-and-Architecture/Code/git/altimetrik.com/bitbucket/projects/DEVOPS/novartis-test
$ git add README.md
gbiswal@AIPL-LT-1025 MINGW64 /d/LuckyLuckyLucky/Altimetrik/DevOps-Platform-Implementation-and-Architecture/Code/git/altimetrik.com/bitbucket/projects/DEVOPS/novartis-test
$ git commit -m "add README"
(master root-commit) 8a6c1e add README
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 README.md
gbiswal@AIPL-LT-1025 MINGW64 /d/LuckyLuckyLucky/Altimetrik/DevOps-Platform-Implementation-and-Architecture/Code/git/altimetrik.com/bitbucket/projects/DEVOPS/novartis-test
$ git push -u origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 215 bytes | 107.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To http://novartis.devops.altimetrik.io:7080/root/novartis-test.git
 * [new branch]      master -> master
branch 'master' set up to track remote branch 'master' from 'origin'.
gbiswal@AIPL-LT-1025 MINGW64 /d/LuckyLuckyLucky/Altimetrik/DevOps-Platform-Implementation-and-Architecture/Code/git/altimetrik.com/bitbucket/projects/DEVOPS/novartis-test
$
```

How to CREATE BRANCH:

we can create branch in 2 ways

1. Manual using git commands
2. Git-Lab dashboard

Using Git commands

Go to bash terminal run below commands

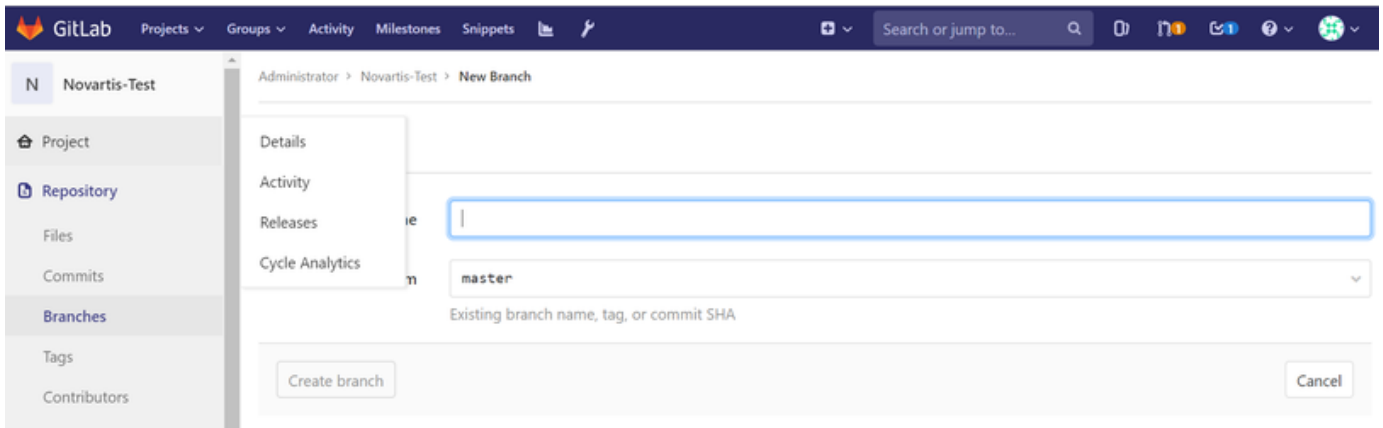
git branch -a

git branch develop

```
gbiswal@AIPL-LT-1025 MINGW64 /d/LuckyLuckyLucky/Altimetrik/DevOps-Platform-Implementation-and-Architecture/Code/git/altimetrik.com/bitbucket/projects/DEVOPS/novartis-test (master)
$ git branch -a
* master
  remotes/origin/master
gbiswal@AIPL-LT-1025 MINGW64 /d/LuckyLuckyLucky/Altimetrik/DevOps-Platform-Implementation-and-Architecture/Code/git/altimetrik.com/bitbucket/projects/DEVOPS/novartis-test (master)
$ git branch develop
gbiswal@AIPL-LT-1025 MINGW64 /d/LuckyLuckyLucky/Altimetrik/DevOps-Platform-Implementation-and-Architecture/Code/git/altimetrik.com/bitbucket/projects/DEVOPS/novartis-test (master)
$
```

Using Git-Lab dashboard

Go to repository > Create branch from here



Sample Hello World programming using Python

1. First lines of code

```
print("hello world")
```

Save the source code in repository in the name of helloworld.py

```
git add .  
git commit -m "Initial Commit"  
git push -u origin master
```

2. Compile the Python code

```
D:\LuckyLuckyLucky\Altimetrik\DevOps-Platform-Implementation-and-Architecture\Code\git\altimetrik.com\bitbucket\projects\DEVOPS\novartis-test>python helloworld.py  
hello world
```