

Table of Contents

- Getting Started and Configuration** 3
- Identification*** 3
- GitHub noreply Email Address 3
- Creating and Downloading Projects*** 3

Getting Started and Configuration

First, we can configure the editor and command-line settings.

```
# Enable automatic command-line highlighting:
git config --global color.ui auto
# Set up the general editor for commits:
git config --global core.editor notepad++
# In older Git versions, the default branch is "master." This command sets
the default branch to "main"—the modern standard—for all newly created
projects.
git config --global init.defaultBranch main
# Lists all your current Git settings, your name, and your email address.
(This is ideal for verifying that your settings are correct.)
git config --list
```

Identification

```
git config --global user.name "İsminiz"
git config --global user.email "mail@adresiniz.com"
```

GitHub noreply Email Address

If we're working on GitHub, we don't want our own email address to appear in our commits. That's why we need to use the **noreply** email address provided by GitHub.

<https://docs.github.com/en/account-and-profile/reference/email-addresses-reference#your-noreply-email-address>

Creating and Downloading Projects

The following command converts a regular folder on your computer into a Git repository. When you run the command, a hidden `.git` directory is created inside the folder. Git stores the entire history there.

```
git init
```

It downloads an existing project from a remote server, such as GitHub or GitLab, to your computer along with its entire history and files.

```
git clone <URL>
```

Taken from [UCH Wiki](#).

From:

<https://wiki.ulascemh.com/> - **UCH**

Permanent link:

<https://wiki.ulascemh.com/doku.php?id=en:cs:devtools:git:config>

Last update: **2026/04/05 13:06**

