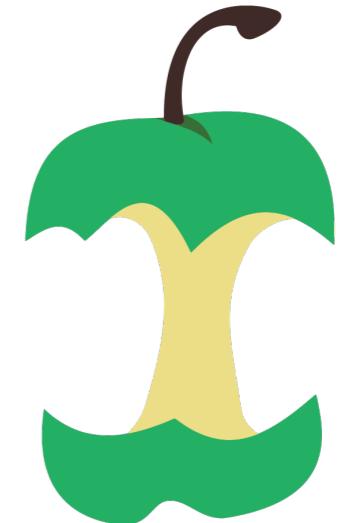


nf-core



Contributing to nf-core

Last updated March 2022

Phil Ewels, Gisela Gabernet, Rike Hanssen

What is nf-core?

Deploy



Stable pipelines



Centralised
configs



List and update
pipelines



Download
for offline use

Participate



Documentation



Slack workspace



Twitter updates



Hackathons

Develop



Starter template



Code guidelines

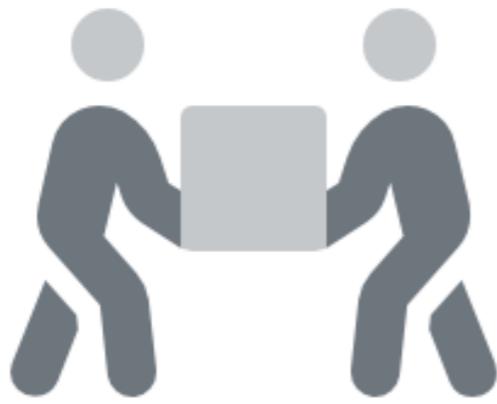


CI code linting
and tests



Helper tools

Contribution guidelines



Develop with the community

Join slack, communicate and contribute together to a pipeline



Cooperate, don't duplicate

One pipeline per analysis type, contribute by adding new tools, new features...

Tutorial



https://nf-co.re/usage/tutorials/nf_core_contributing_overview

- 1 Installation
- 2 Creating
- 3 Testing

- 4 Modules
- 5 Releasing

Installation

Helper tools

nf-core/ **tools**



```
pip install nf-core
```



BIOCONDA®

```
conda install -c bioconda nf-core
```

Helper tools

nf-core --help

```
  ,--./,--.
  /,-._.-~\
   } {
 \`.-_,-`-
  .-,--,'

NF-CORE [--version | --verbose | --log-file | --help]
 nf-core/tools version 2.3 - https://nf-co.re

Usage: nf-core [OPTIONS] COMMAND [ARGS]...

nf-core/tools provides a set of helper tools for use with nf-core Nextflow
pipelines.
It is designed for both end-users running pipelines and also developers
creating new pipelines.

Options
  --version           Show the version and exit.
  --verbose    -v      Print verbose output to the console.
  --log-file   -l <filename> Save a verbose log to a file.
  --help       -h      Show this message and exit.

Commands for users
  list      List available nf-core pipelines with local info.
  launch    Launch a pipeline using a web GUI or command line prompts.
  download  Download a pipeline, nf-core/configs and pipeline singularity
            images.
  licences  List software licences for a given workflow.

Commands for developers
  create    Create a new pipeline using the nf-core template.
  lint      Check pipeline code against nf-core guidelines.
  modules   Commands to manage Nextflow DSL2 modules (tool wrappers).
  schema    Suite of tools for developers to manage pipeline schema.
  bump-version Update nf-core pipeline version number.
  sync      Sync a pipeline TEMPLATE branch with the nf-core template.
```

Helper tools

nf-core list



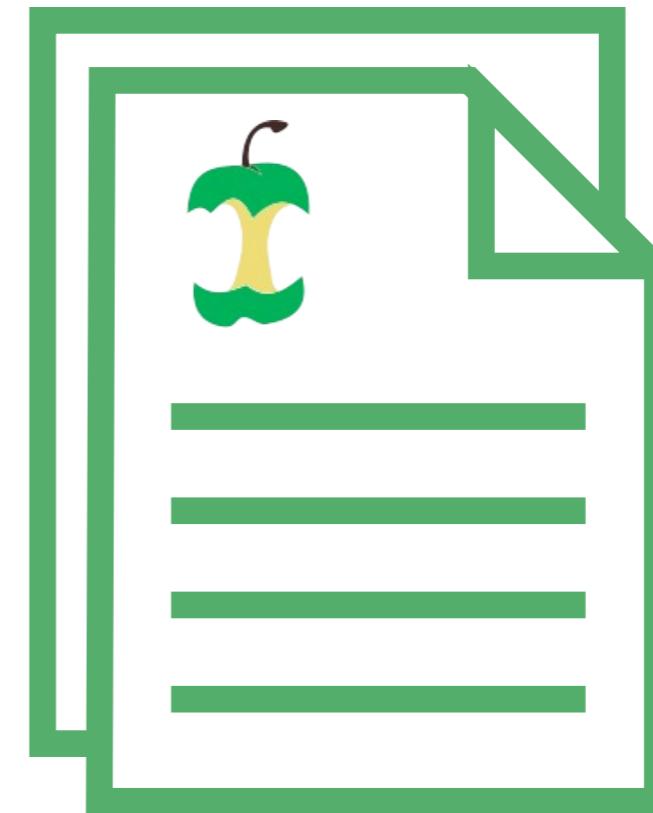
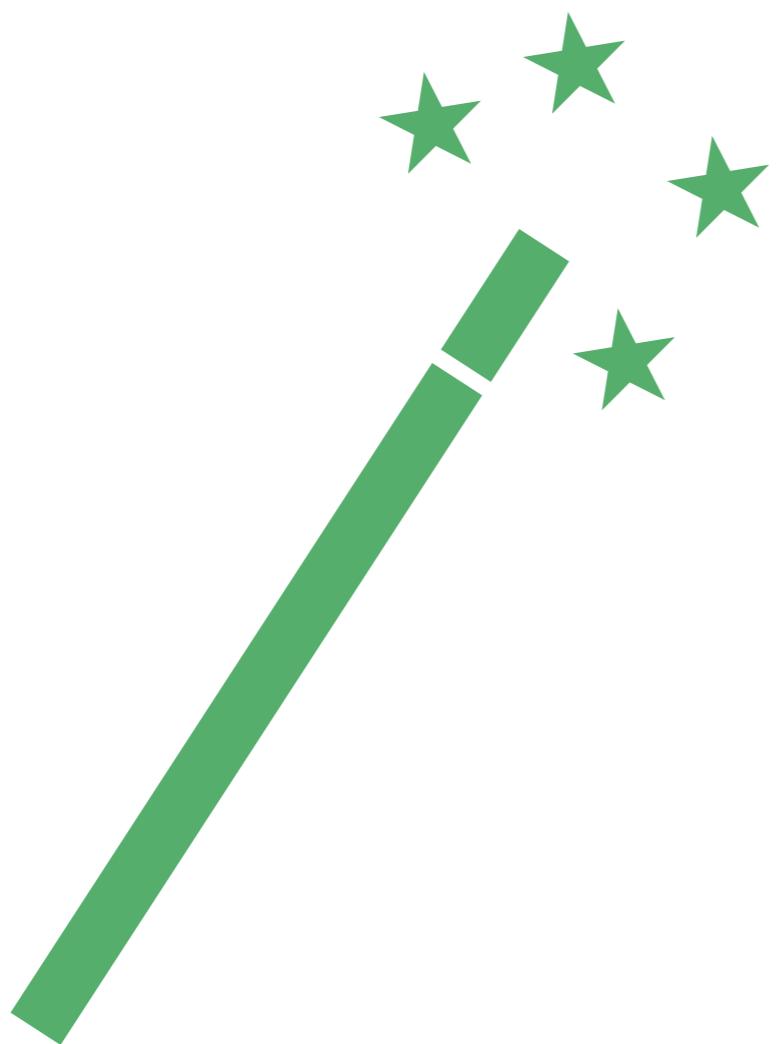
nf-core/tools version 2.3 - <https://nf-co.re>

Pipeline Name	Stars	Latest Release	Released	Last Pulled	Have latest release?
<code>rnaseq</code>	433	3.6	2 weeks ago	4 months ago	No (v3.4)
<code>viralrecon</code>	61	2.4.1	2 weeks ago	3 weeks ago	No (v2.3.1)
<code>ampliseq</code>	82	2.2.0	1 months ago	-	-
<code>eager</code>	61	2.4.2	2 months ago	-	-
<code>cutandrun</code>	24	1.1	2 months ago	-	-
<code>mhcquant</code>	19	2.2.0	2 months ago	-	-
<code>epitopeprep...</code>	18	2.0.0	3 months ago	2 months ago	Yes (v2.0.0)
<code>fetchngs</code>	52	1.5	4 months ago	-	-
<code>nanoseq</code>	68	2.0.1	4 months ago	-	-
<code>mag</code>	81	2.1.1	4 months ago	4 months ago	Yes (v2.1.1)
<code>bacass</code>	33	2.0.0	7 months ago	-	-
<code>airrflow</code>	17	2.0.0	8 months ago	-	-

Creating pipelines



nf-core create



Template

nf-core create

```
MEET -- ( ) [ ] \ / , - . _ . ~ \ } { \ ' - . _ , - ' - , ' . _ , - , ' nf-core/tools version 2.3 - https://nf-co.re

Workflow Name: mypipeline
Description: This is a demo pipeline for the tutorial
Author: Gisela Gabernet
INFO Creating new nf-core pipeline: 'nf-core/mypipeline'
INFO Initialising pipeline git repository
INFO Done. Remember to add a remote and push to GitHub:
cd /Users/gisela/Projects/Pipelines/nf-core-mypipeline
git remote add origin git@github.com:USERNAME/REPO_NAME.git
git push --all origin
INFO This will also push your newly created dev branch and the
TEMPLATE branch for syncing.
INFO !!!!!! IMPORTANT !!!!!!

If you are interested in adding your pipeline to the nf-core
community,
PLEASE COME AND TALK TO US IN THE NF-CORE SLACK BEFORE WRITING ANY
CODE!

Please read:
https://nf-co.re/developers/adding_pipelines#join-the-community
```

Template

```
usage.md
49 NXF_OPTS=' -Xms1g -Xmx4g '
50 ``
51 |
52 <-- TODO nf-core: Document required command line parameters to run the pipeline-->
53 |
54 ## Running the pipeline
55 The typical command for running the pipeline is as follows:
56 |
57 ````bash
58 nextflow run nf-core/mypipeline --reads '*_R{1,2}.fastq.gz' -profile docker
59 ````

60 |
61 This will launch the pipeline with the `docker` configuration profile. See below for more
• information about profiles.
62 |
63 Note that the pipeline will create the following files in your working directory:
64 |
65 ````bash
66 work          # Directory containing the nextflow working files
67 results        # Finished results (configurable, see below)
68 .nextflow_log # Log file from Nextflow
69 # Other nextflow hidden files, eg. history of pipeline runs and old logs.
70 ````
```

Template

Template

```
├── CHANGELOG.md  
├── CITATIONS.md  
├── CODE_OF_CONDUCT.md  
└── LICENSE  
├── README.md  
└── assets  
    └── ...  
├── bin  
    └── ...  
├── conf  
    └── ...  
└── docs  
    └── ...  
├── lib  
    └── ...  
├── main.nf  
└── modules  
    ├── local  
    └── nf-core  
├── modules.json  
├── nextflow.config  
├── nextflow_schema.json  
└── subworkflows  
    └── local  
└── workflows  
    └── pipeline.nf
```

- 12 directories
- 32 files
- + > 40 hidden files

Template

```
├── CHANGELOG.md  
├── CITATIONS.md  
├── CODE_OF_CONDUCT.md  
└── LICENSE  
├── README.md  
├── assets  
│   └── ...  
├── bin  
│   └── ...  
├── conf  
│   └── ...  
├── docs  
│   └── ...  
└── lib  
    └── ...  
    └── main.nf  
    └── modules  
        └── local  
            └── nf-core  
    └── modules.json  
    └── nextflow.config  
    └── nextflow_schema.json  
    └── subworkflows  
        └── local  
    └── workflows  
        └── pipeline.nf
```

Nextflow code

DSL2 brings modularity

A pipeline can contain:

- modules
- subworkflows
- workflows

Template

```
├── CHANGELOG.md  
├── CITATIONS.md  
├── CODE_OF_CONDUCT.md  
└── LICENSE  
├── README.md  
├── assets  
│   └── ...  
├── bin  
│   └── ...  
├── conf  
│   └── ...  
├── docs  
│   └── ...  
└── lib  
    ├── NfcoreSchema.groovy  
    ├── NfcoreTemplate.groovy  
    ├── Utils.groovy  
    ├── pipeline.groovy  
    ├── WorkflowMain.groovy  
    └── nfcore_external_java_deps.jar  
└── main.nf  
└── modules  
└── modules.json  
└── nextflow.config  
└── nextflow_schema.json  
└── subworkflows  
└── workflows
```

Groovy classes and functions used by the pipeline

Template

```
├── CHANGELOG.md  
├── CITATIONS.md  
├── CODE_OF_CONDUCT.md  
└── LICENSE  
└── README.md  
└── assets  
    ├── email_template.html  
    ├── email_template.txt  
    ├── multiqc_config.yaml  
    ├── nf-core-pipeline_logo.png  
    ├── samplesheet.csv  
    ├── schema_input.json  
    └── sendmail_template.txt  
└── bin  
    └── check_samplesheet.py  
├── conf  
└── docs  
└── lib  
└── main.nf  
└── modules  
└── modules.json  
└── nextflow.config  
└── nextflow_schema.json  
└── subworkflows  
└── workflows
```

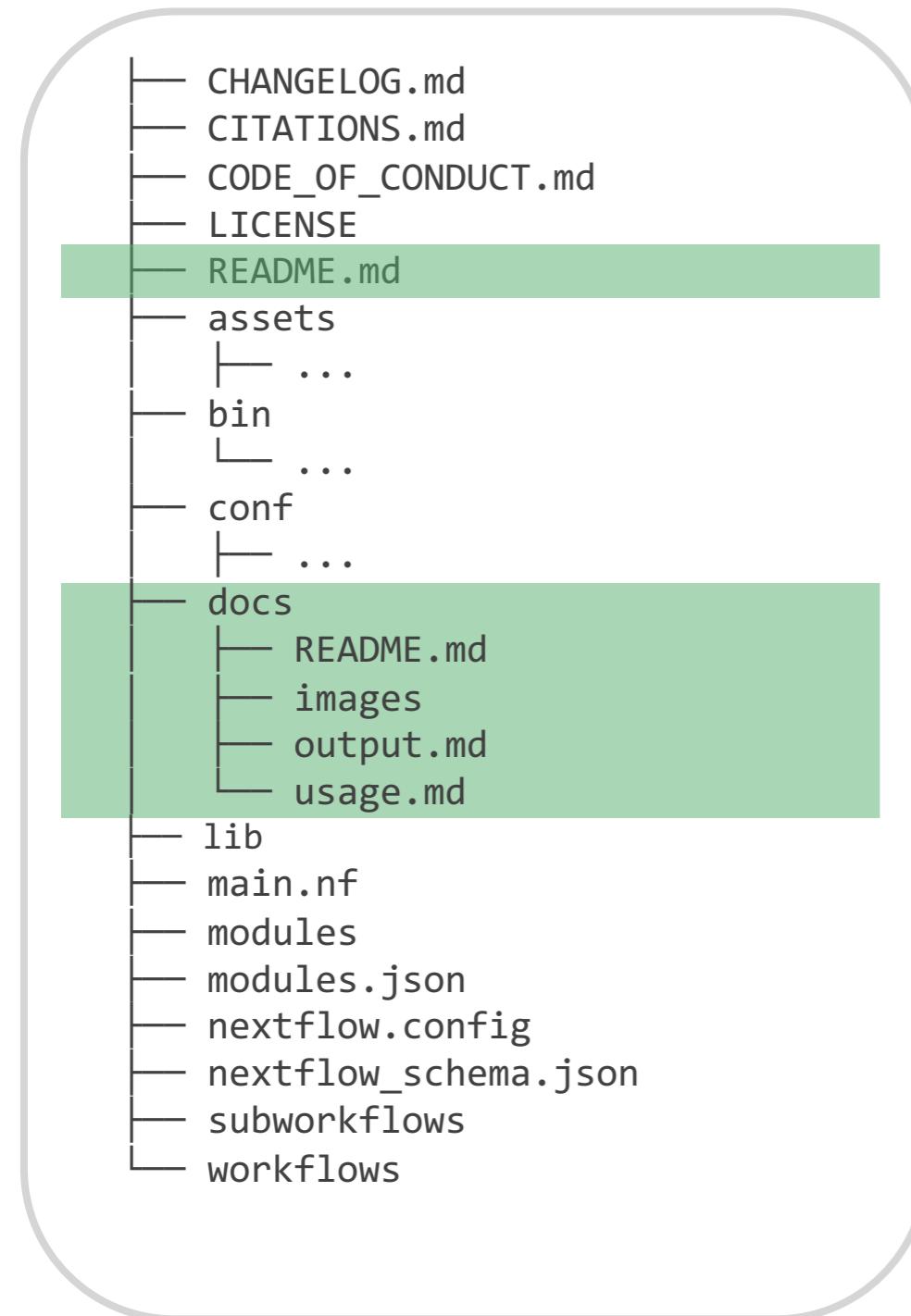
bin/

- **Executable scripts**
- Added to the path by Nextflow

assets/

- Email template
- MultiQC config
- Pipeline logo
- Input file schema

Template



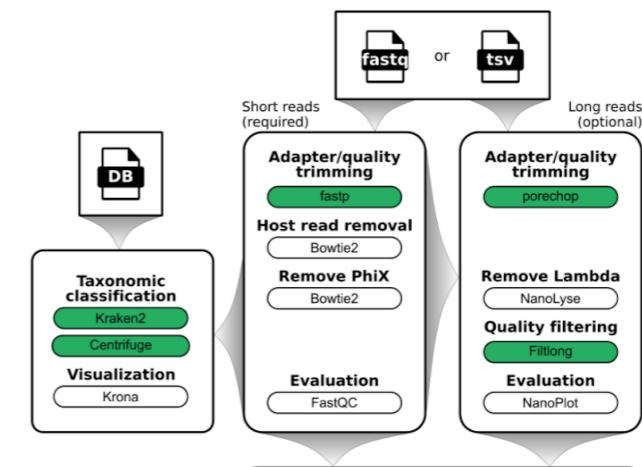
Documentation

- rendered on the website

Introduction AWS Results Usage docs Parameter docs Output docs

Introduction

nf-core/mag is a bioinformatics best-practise analysis pipeline for assembly, binning, and annotation of metagenomes.



Template

```
└── CHANGELOG.md  
└── CITATIONS.md  
└── CODE_OF_CONDUCT.md  
└── LICENSE  
└── README.md  
└── assets  
    └── ...  
└── bin  
    └── ...  
└── conf  
    └── ...  
└── docs  
    ├── README.md  
    ├── images  
    ├── output.md  
    └── usage.md  
└── lib  
└── main.nf  
└── modules  
└── modules.json  
└── nextflow.config  
└── nextflow_schema.json └── subworkflows  
└── workflows
```

Parameter documentation

- rendered from json

[Introduction](#) [aws Results](#) [Usage docs](#) **Parameter docs** [Output docs](#) [|](#)

Parameters

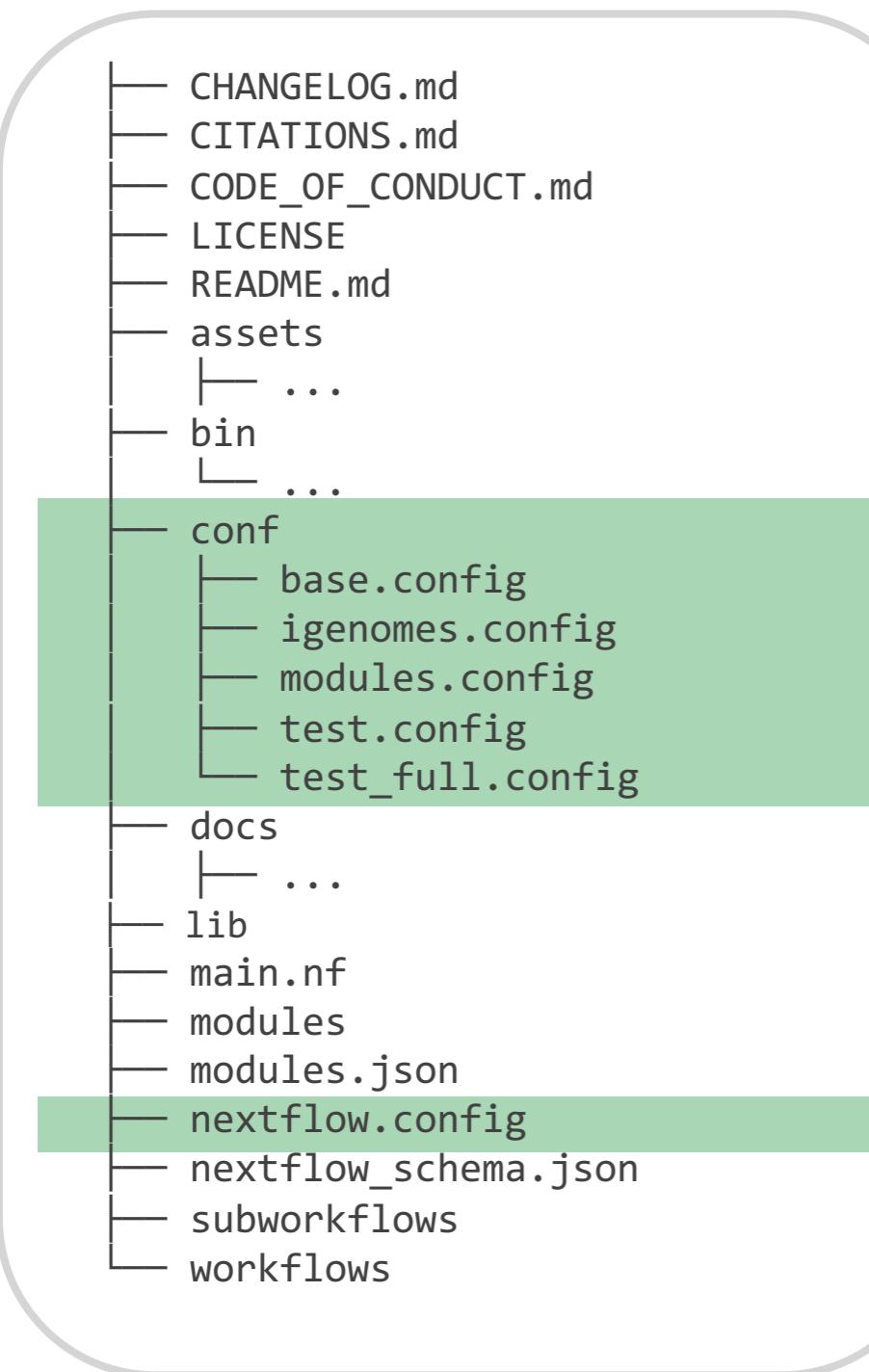
> Input/output options

Define where the pipeline should find input data and save output data.

<code>--input</code>	Input FastQ files or TSV file.	? Help
<code>--single_end</code>	Specifies that the input is single-end reads.	? Help
<code>--outdir</code>	The output directory where the results will be saved. <small>default: './results'</small>	
<code>--email</code>	Email address for completion summary.	? Help

nf-core schema build

Template



Configuration

`nextflow.config` imports:



Default 'base' config (always loaded)



Core profiles (eg. docker, conda, test)

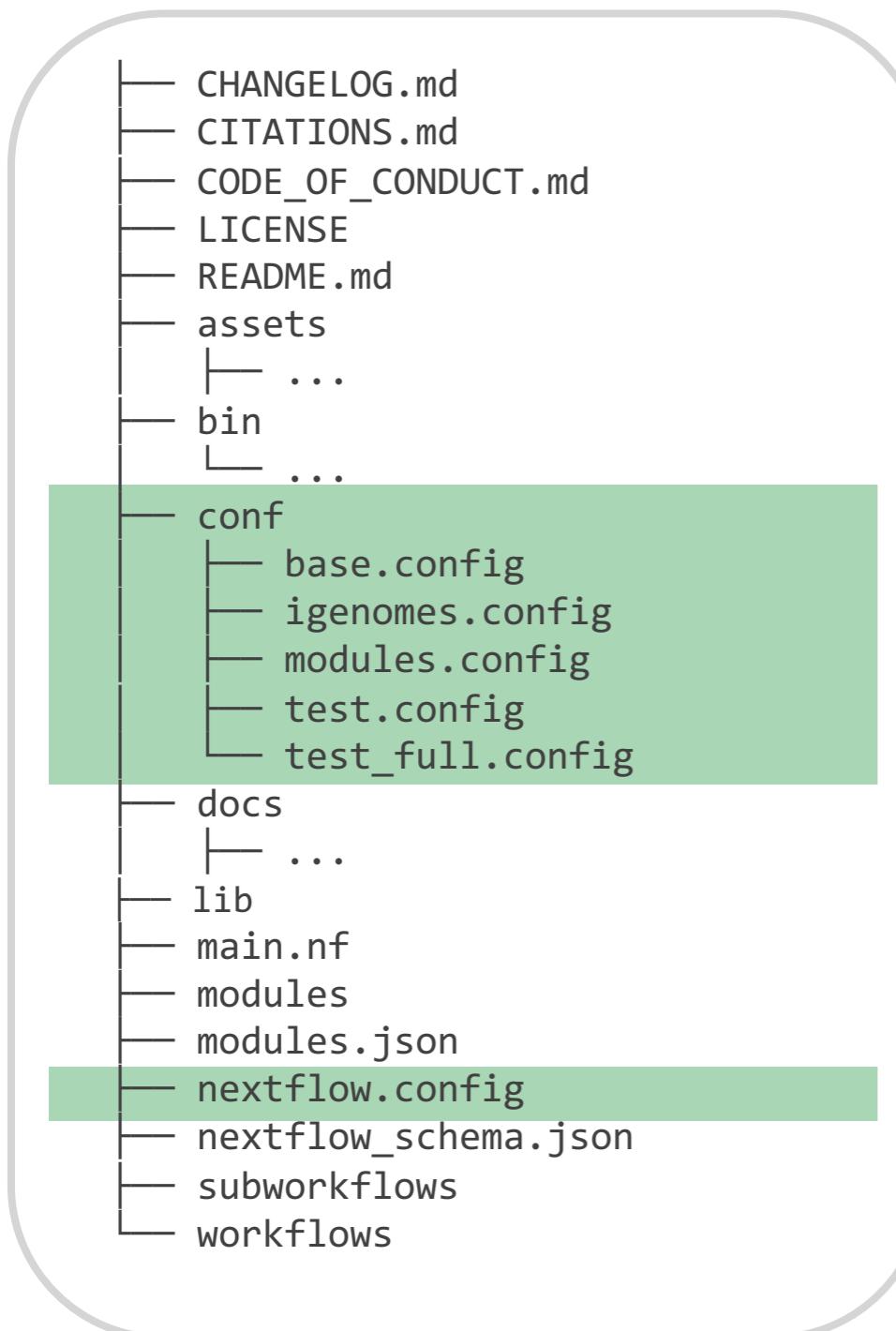


Server profiles (nf-core/configs)



Your local config files

Template



Configuration

nextflow.config imports:



Test profile



Full test profile

Run with GitHub actions:



Template

the hidden files:

```
.editorconfig  
.github  
|   .dockstore.yml  
|   CONTRIBUTING.md  
|   ISSUE_TEMPLATE  
|       bug_report.yml  
|       config.yml  
|       feature_request.yml  
|  
|   PULL_REQUEST_TEMPLATE.md  
|  
|   workflows  
|       awsfulltest.yml  
|       awstest.yml  
|       branch.yml  
|       ci.yml  
|       linting.yml  
|       linting_comment.yml  
  
.gitignore  
.markdownlint.yml
```

GitHub actions workflows

- AWS full pipeline tests
- AWS pipeline tests
- Branch protection
- CI tests
- Linting tests
- Linting comment



Forks & branches



git

Forks & branches



Branch

Merge

Forks & branches



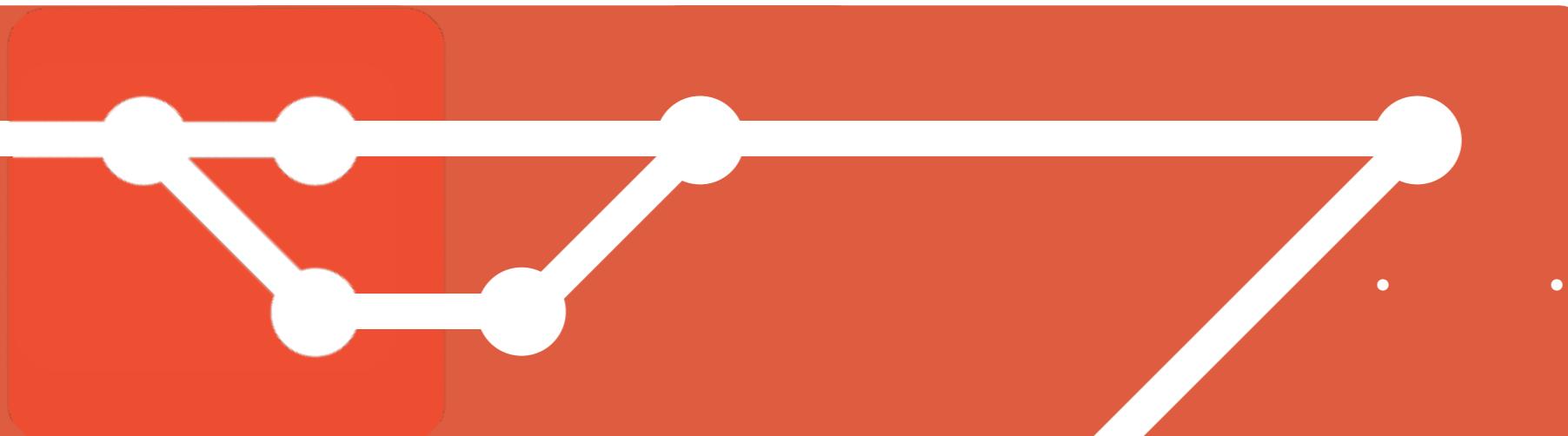
@nf-core

: Fork



@ewels

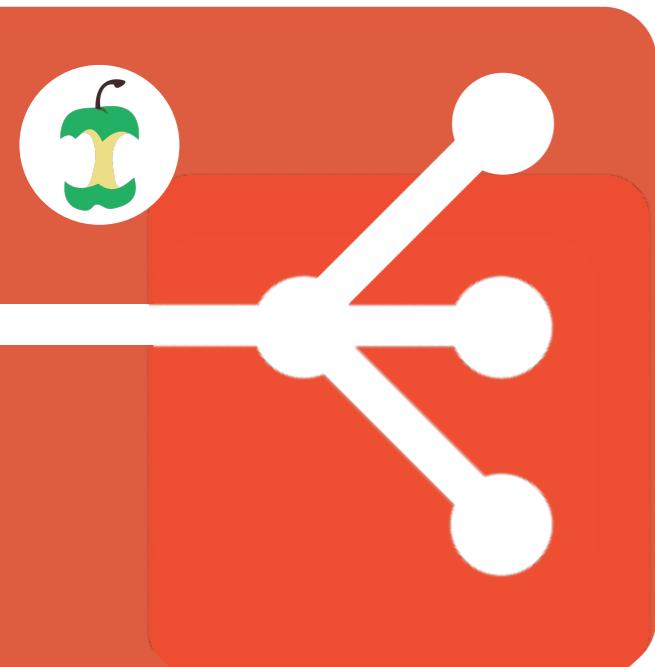
Forks & branches



Pull Request

Merge

Forks & branches

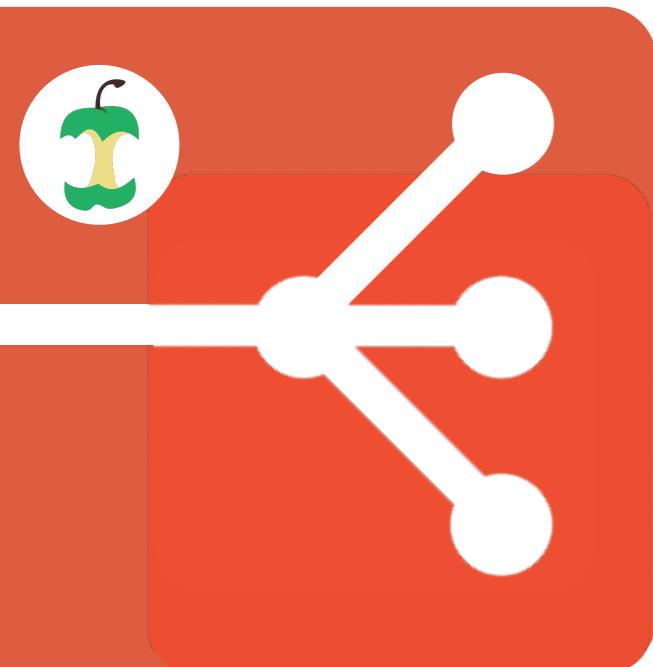


dev

master

TEMPLATE

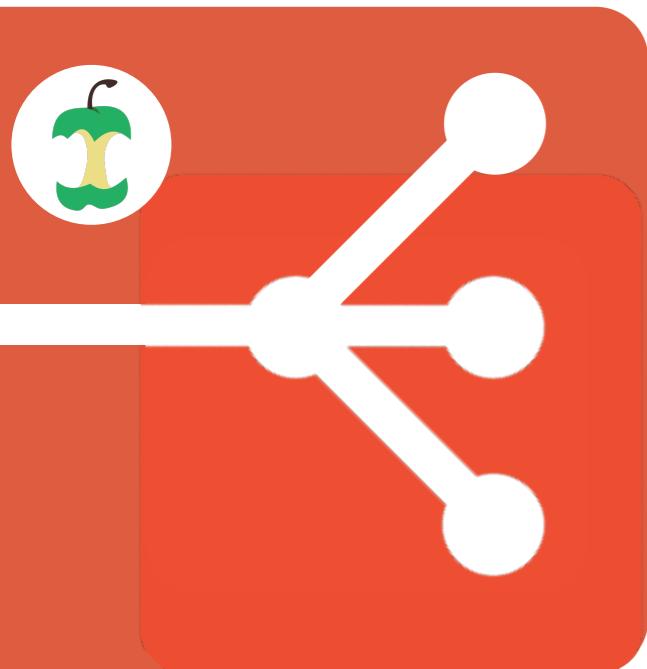
Forks & branches



dev
master
TEMPLATE

Stable releases only

Forks & branches



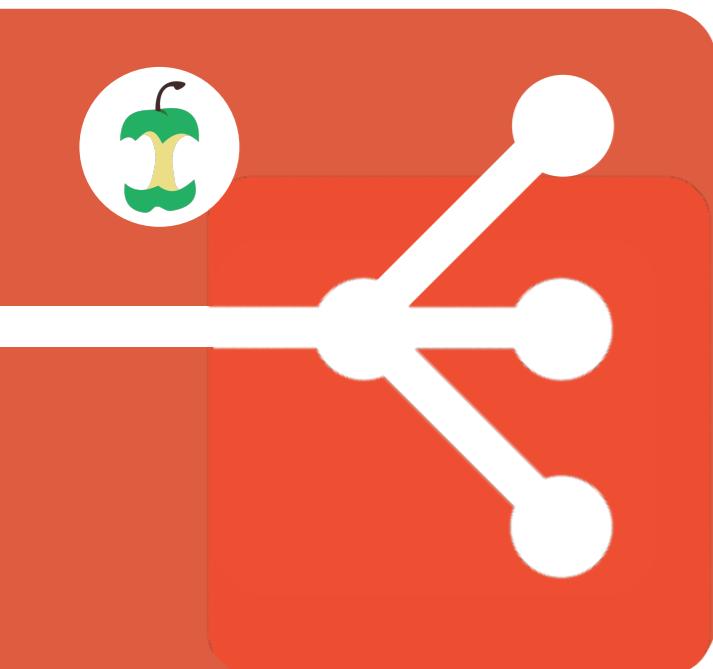
dev

master

TEMPLATE

Used by @nf-core-bot
to keep in sync

Forks & branches



dev
master
TEMPLATE

All development code
(and all pull requests)

Forks & branches

 REMEMBER!



PRs must be against the dev branch



Changelog must be updated



PRs need to be reviewed

Testing pipelines



Lint tests

nf-core lint

```
          ,--./,-.  
 /,-._.-~\  
 } {  
\`-._,-`-,  
`-.,-,'  
  
Running pipeline tests [#####-----] 44% 'check_nextflow_config'  
ERROR: Found test failures in 'check_nextflow_config', halting lint run.
```

INFO: =====

LINTING RESULTS

=====

46 tests passed 0 tests had warnings 1 tests failed

ERROR: Test Failures:

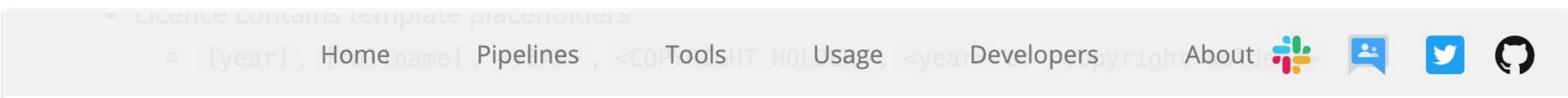
<http://nf-co.re/errors#4>: Config variable (incorrectly) found: params.container

ERROR: Sorry, some tests failed - exiting with a non-zero error code...



Lint tests

<https://nf-co.re/errors>



Error #4 - Nextflow config check failed

nf-core pipelines are required to be configured with a minimal set of variable names. This test fails or throws warnings if required variables are not set.

Note: These config variables must be set in `nextflow.config` or another config file imported from there.
Any variables set in nextflow script files (eg. `main.nf`) are not checked and will be assumed to be missing.

The following variables fail the test if missing:

- `params.outdir`
 - A directory in which all pipeline results should be saved
- `manifest.name`
 - The pipeline name. Should begin with `nf-core/`
- `manifest.description`
 - A description of the pipeline
- `manifest.version`
 - The version of this pipeline. This should correspond to a GitHub release.

Test data

```
nextflow run nf-core/rnaseq -profile test,docker
```

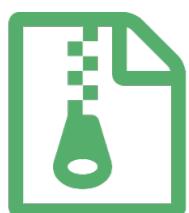
nf-core/ 
test-datasets

Test data

nf-core/ **test-datasets**



Make a new branch for your pipeline



Add a really (really!) tiny dataset



Set up the test config to use raw URLs

Exercises 2, 3

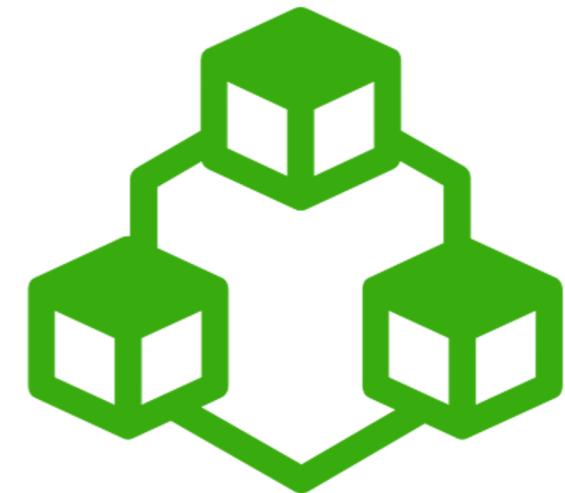
Exercise 2 (creating pipelines)

- Make a new pipeline using the template
- Update the readme file to fill in the TODO statements
- Add a new process to the pipeline in `main.nf`
- Add the new software dependencies from this process in to `environment.yaml`

Exercise 3 (testing pipelines)

- Run `nf-core lint` on your pipeline and make note of any test warnings / failures
- Read up on one or two of the linting rules on the nf-core website and see if you can fix some.
- Take a look at `conf/test.config` and switch the test data for another dataset on `nf-core/test_data`

nf-core modules



modules

nf-core modules --help

```
[NF] nf-core modules --help
```

```
nf-core/tools version 2.3 - https://nf-core.org
```

Usage: nf-core modules [OPTIONS] COMMAND [ARGS]...

Commands to manage Nextflow DSL2 modules (tool wrappers).

— Options —

```
--github-repository -g TEXT GitHub repository hosting modules.  
--branch -b TEXT Branch of GitHub repository hosting modules.  
--help -h Show this message and exit.
```

— For pipelines —

```
list List modules in a local pipeline or remote repository.  
info Show developer usage information about a given module.  
install Install DSL2 modules within a pipeline.  
update Update DSL2 modules within a pipeline.  
remove Remove a module from a pipeline.
```

— Developing new modules —

```
create Create a new DSL2 module from the nf-core template.  
create-test-yml Auto-generate a test.yml file for a new module.  
lint Lint one or more modules in a directory.  
bump-versions Bump versions for one or more modules in a clone of the  
nf-core/modules repo.
```

modules

nf-core modules list

```
[nf-core] nf-core/modules [master] ⚡
```

nf-core/tools version 2.3 - <https://nf-core.org>

Usage: nf-core modules list [OPTIONS] COMMAND [ARGS]...

List modules in a local pipeline or remote repository.

Options

--help -h Show this message and exit.

Commands

local List modules installed locally in a pipeline

remote List modules in a remote GitHub repo (e.g *nf-core/modules*).

modules

Modules

Browse the **440** modules that are currently available as part of nf-core.

Available Modules

Modules are the building stones of all DSL2 nf-core blocks. You can find more info , if you would like to write your own module.

Search modules

fasta
bam
reference
index

69
64
34
32
21

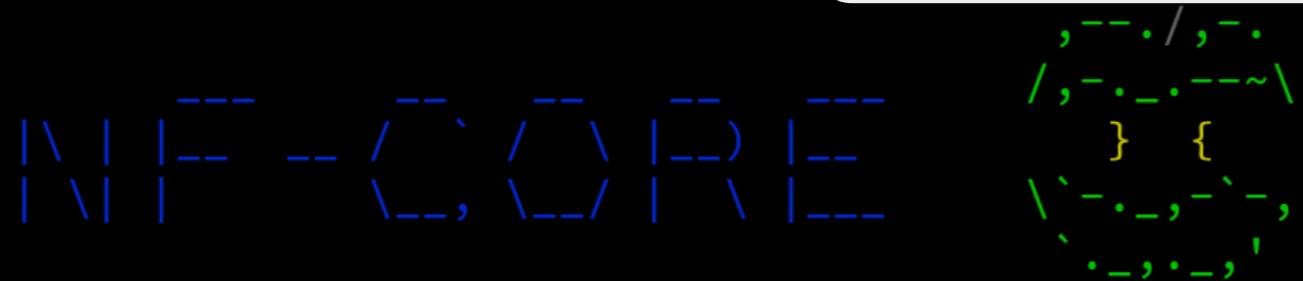
[abacas](#)

[genome](#) [assembly](#) [contiguate](#)

contiguate draft genome assembly

modules

nf-core modules install



nf-core/tools version 2.3 - <https://nf-co.re>

? Tool name: bwa/mem

INFO Installing 'bwa/mem'

INFO Downloaded 2 files to ./modules/nf-core/modules/bwa/mem

modules

```
INFO -- (NF-CORE) -- /,-. /,-. } \,-. ,--,'
```

nf-core modules create

```
nf-core/tools version 2.3 - https://nf-co.re

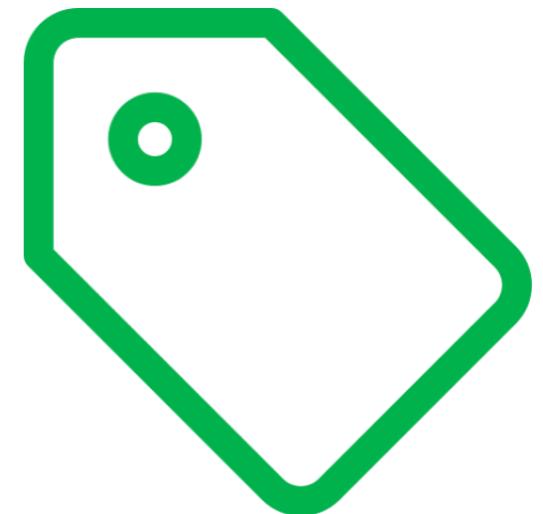
INFO Repository type: pipeline
INFO Press enter to use default values (shown in brackets) or type your
own responses. ctrl+click underlined text to open links.
Name of tool/subtool: fastqc
INFO Using Bioconda package: 'bioconda::fastqc=0.11.9'
INFO Using Docker container:
'quay.io/biocontainers/fastqc:0.11.9--hdfd78af_1'
INFO Using Singularity container: 'https://depot.galaxyproject.org/singularity/fastqc:0.11.9--hdfd78af\_1'
GitHub Username: (@ggabernet):
INFO Provide an appropriate resource label for the process, taken from
the nf-core pipeline template.
For example: process_low, process_medium, process_high,
process_long
? Process resource label: process_low
INFO Where applicable all sample-specific information e.g. 'id', 'single_end', 'read_group' MUST be provided as an input via a
Groovy Map called 'meta'. This information may not be required in
some instances, for example indexing reference genome files.
Will the module require a meta map of sample information? [y/n] (y): y
INFO Created / edited following files:
./modules/local/fastqc.nf
```

Exercise 4

Exercise 4 (modules)

- Use the pipeline you created in Exercise 2 and add an already existing nf-core module (e.g. trimgalore).
- Connect the module to the main pipeline workflow.

Releasing pipelines



First release

https://nf-co.re/developers/adding_pipelines



Tell us about it: **#new-pipelines**



Join the @nf-core organisation



Fork your repo to nf-core



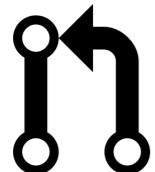
Set up Zenodo

First release

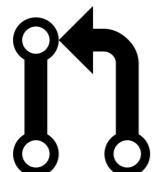
https://nf-co.re/developers/adding_pipelines



Make sure that all tests are passing



Community review: "Pseudo PR"



Open PR from dev to master



Create release on GitHub

Version numbers

Semantic versioning:

1.4.3

Major - results not backwards compatible

Minor - new feature(s)

Patch - bug fix

nf-core bump-version 1.4.3

nf-core bump-version --nextflow 19.07.0

Help!



<https://nf-co.re/join>

Template updates

 nf-core / rnaseq
forked from ewels/nf-core-rnaseq

Unwatch ▾ 38 Star 127 Fork 136

Code Issues 32 Pull requests 6 Actions Security Insights Settings

Important pipeline nf-core update! (version 1.6) #194

Merged drpatelh merged 1 commit into dev from TEMPLATE on 12 Jun

Conversation 7 Commits 1 Checks 0 Files changed 22 +64 -375

 nf-core-bot commented on 9 Apr

Member + ...

Some important changes have been made in the nf-core pipelines templates.
Please make sure to merge this in ASAP and make a new minor release of your pipeline.
Follow the link [nf-core/tools](#)

 Update nf-core pipeline template. X 55d617e

 ewels referenced this pull request on 11 Jun

Minor edit to installation.md #225 Merged

Reviewers: No reviews

Assignees: No one—assign yourself

Labels: None yet

Projects: None yet

Milestone

Exercise 5

Exercise 5 (releasing pipelines)

- Use nf-core bump-version to update the required version of Nextflow in your pipeline
- Bump your pipeline's version to 1.0, ready for its first release!
- Make sure that you're signed up to the nf-core slack (<https://nf-co.re/join>) and drop us a line about your latest and greatest pipeline plans!
- Ask to be a member of the nf-core GitHub organisation by commenting on this GitHub issue:
<https://github.com/nf-core/nf-co.re/issues/3>
- If you're a twitter user, make sure to follow the @nf_core twitter account