Trimming and Filtering

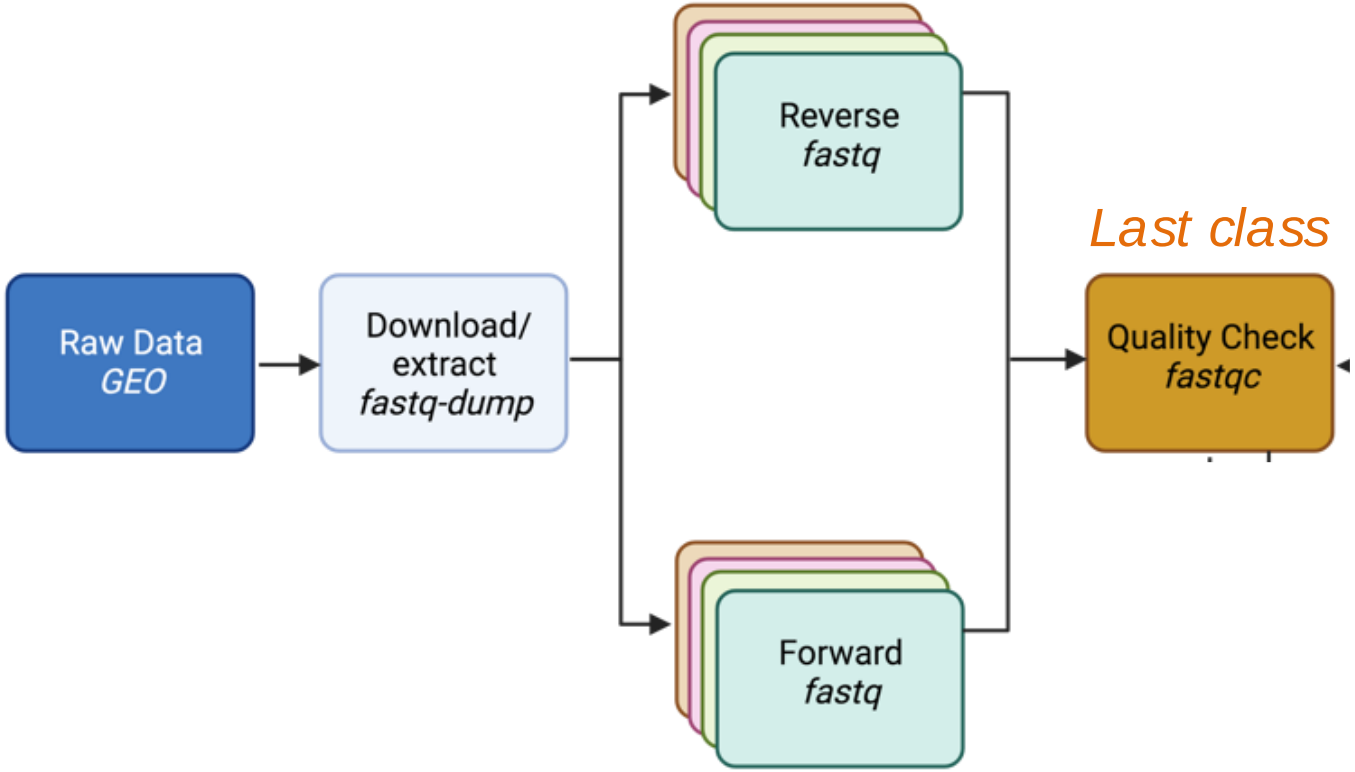
Dr. Princess Rodriguez

2025-01-29

Learning Objectives

- Understand what scenarios warrant trimming
- Be able to clean FASTQ reads using Trimmomatic if required

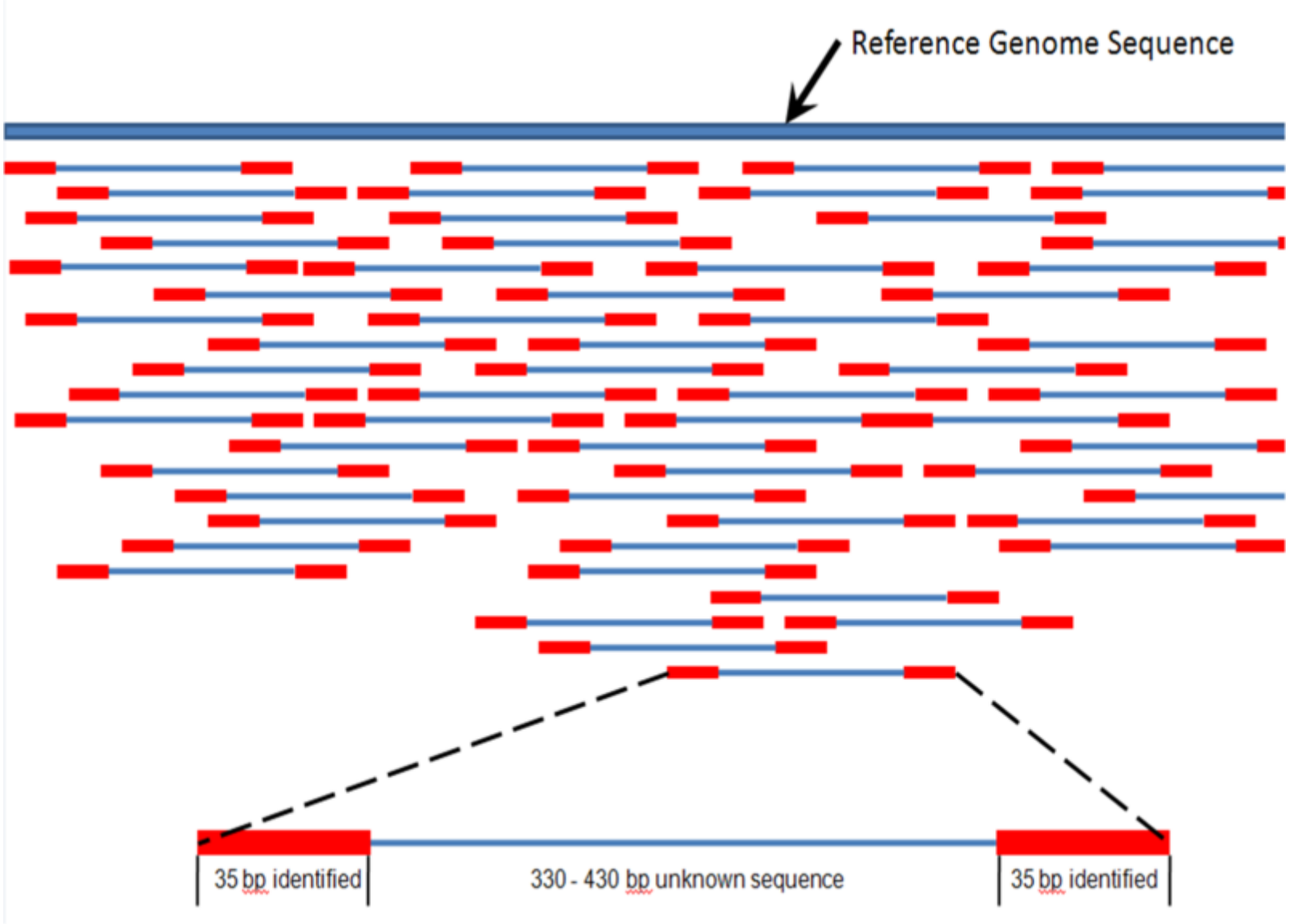
Overview


Raw Data
GEO
Download/
extract
fastq-dump
Reverse
fastq
Forward
fastq
Last class
Quality Check
fastqc

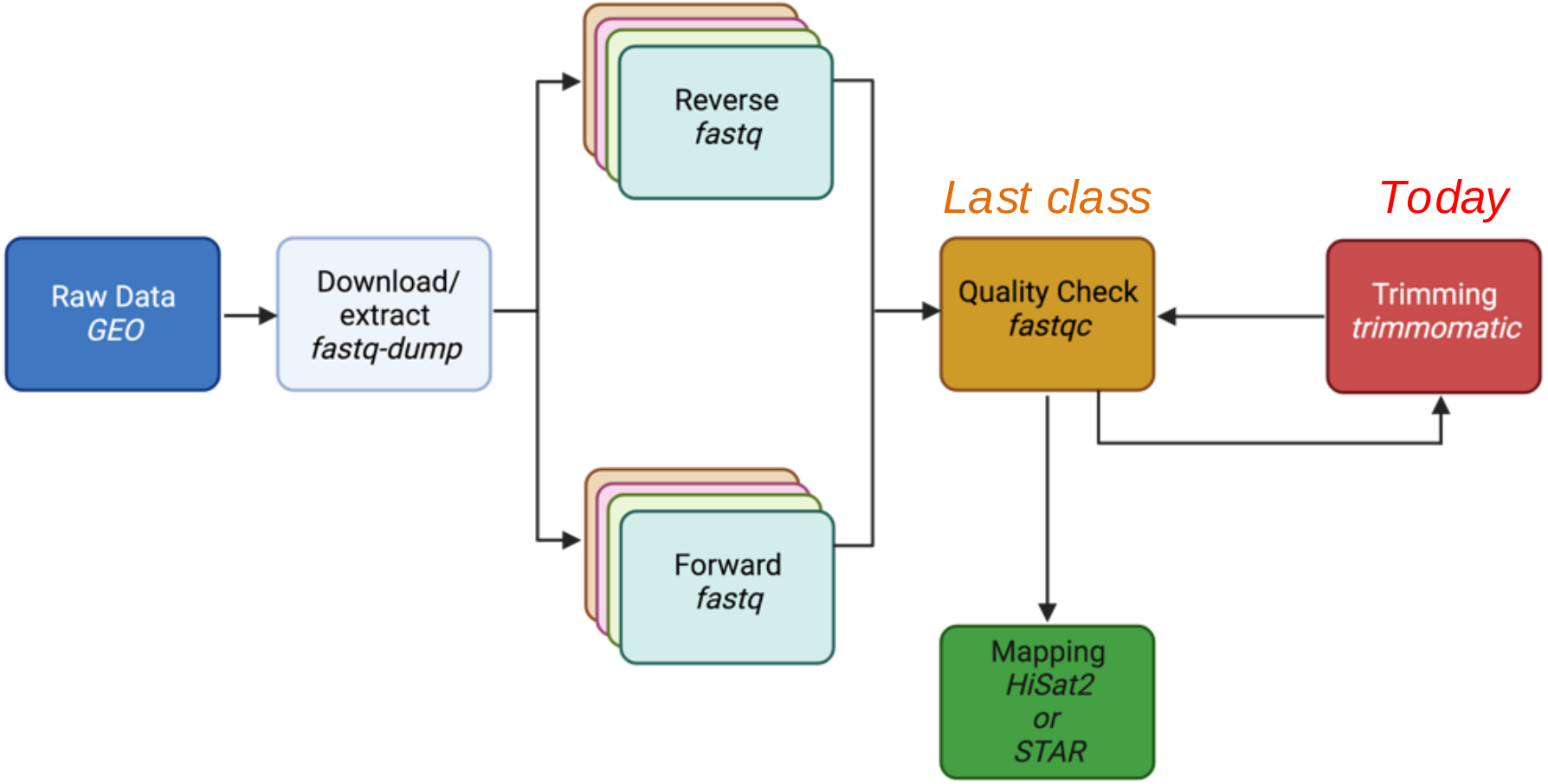
Interpreting the HTML report

- Within the report, a summary of all of the modules is given on the left-hand side.
- Do not take the **yellow "WARNING"s** and **red "FAIL"s as** ***"this sample is not usable"***; they should be interpreted as flags!

Introduction



Overview

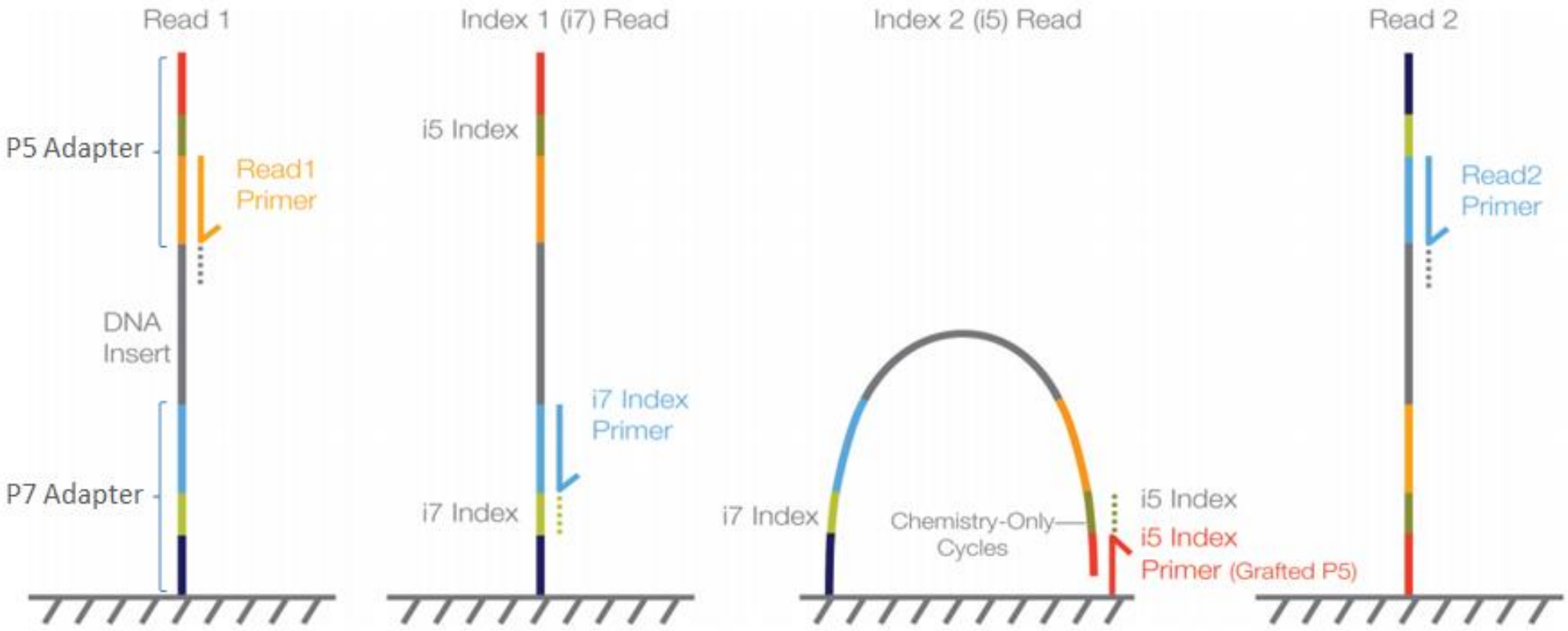

Raw Data
GEO
Download/
extract
fastq-dump
Reverse
fastq
Forward
fastq
Last class
Today
Quality Check
fastqc
Trimming
trimmomatic
Mapping
HiSat2
or
STAR

What to trim?

The types of unwanted information can include one or more of the following:

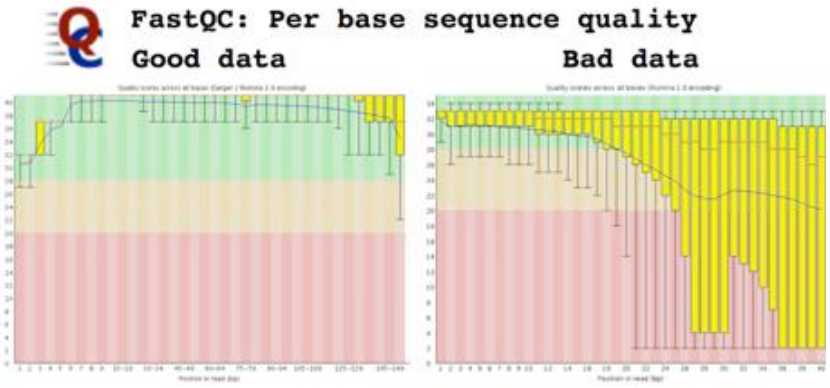
1. leftover adapter sequences
2. known contaminants (strings of As/Ts, other sequences)
3. poor quality bases

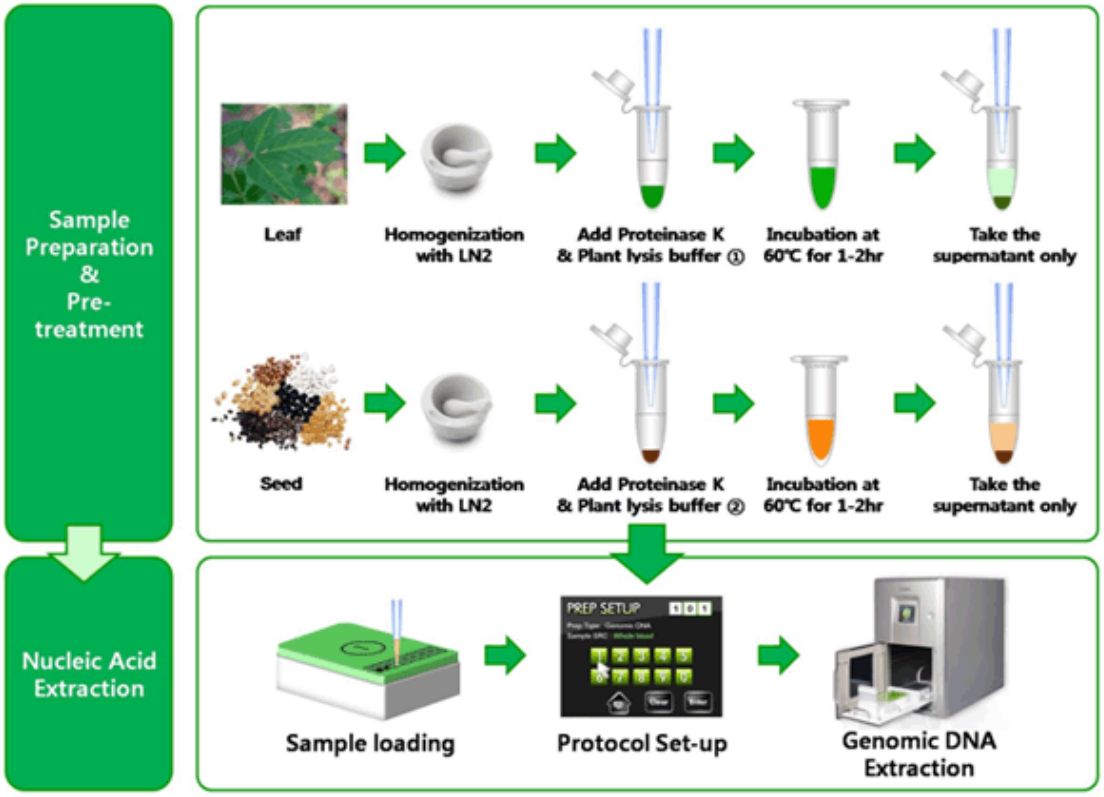
Where does this “unwanted information” come from?



Adapters do not map to the genome

Where does this “unwanted information” come from?





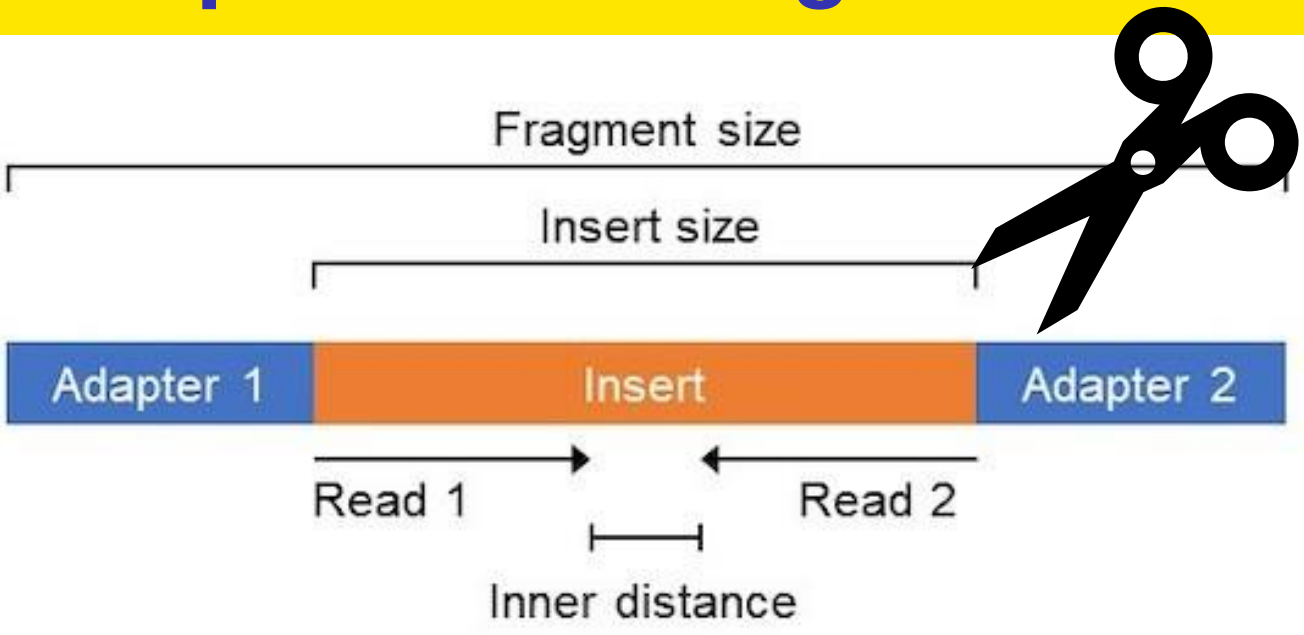
Low quality reads will not map to the genome

Steps to take when considering trimming

Step 1: Quality Control

The first step in trimming RNA-seq data is to assess the quality of the raw reads. This can be done using software such as FastQC, which generates a report. If the data is of poor quality, it may need to be *re-sequenced* or excluded from further analysis.

Step 2: Adapter Trimming



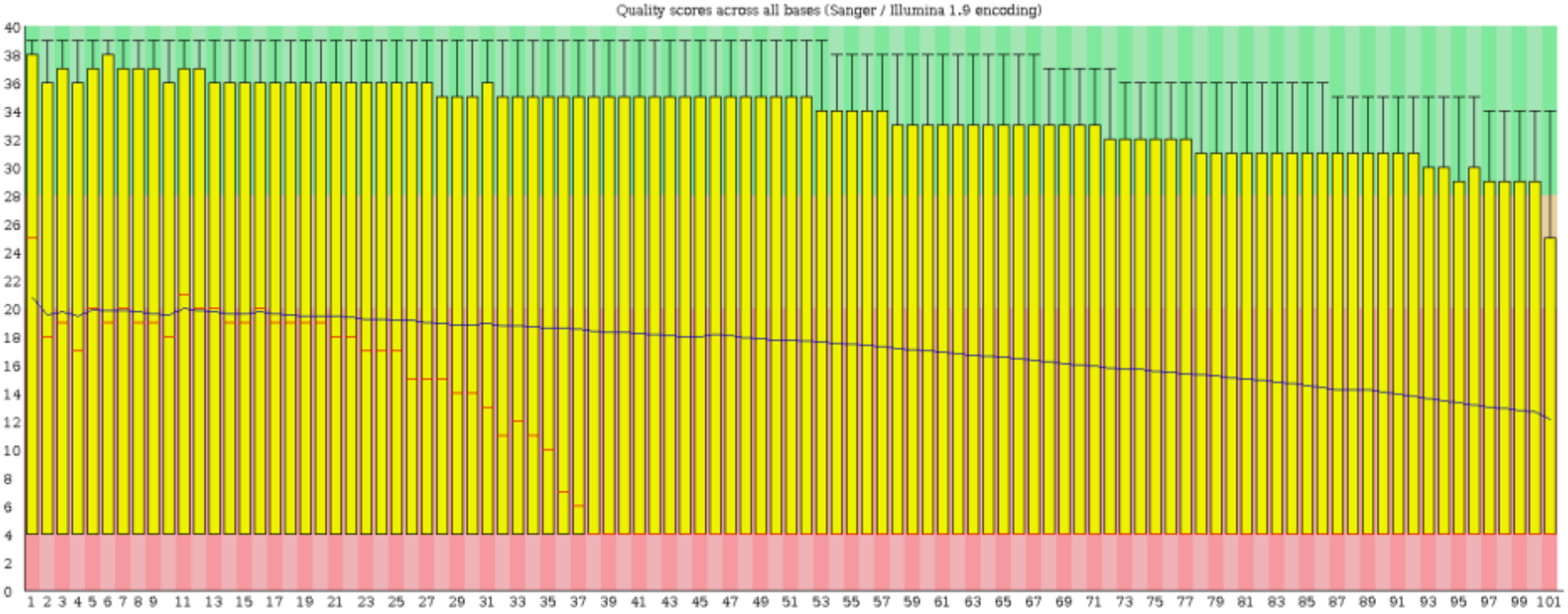
Insert

Insert

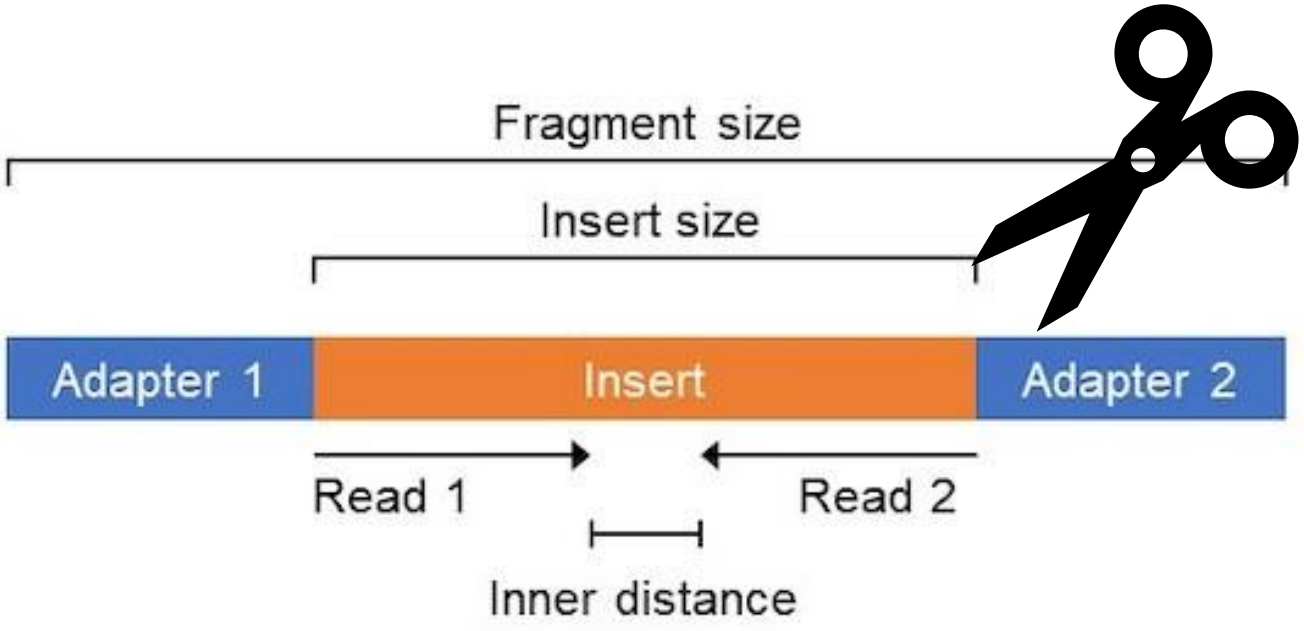
Insert

Step 2b: Quality Trimming

During trimming, we could also remove low-quality reads by setting a *minimum quality threshold* and removing any bases that fall below this threshold.



Step 2c: Short read filtering



55bp

25bp

Tools for Trimming

There are a number of tools that can be used for read trimming, some include:

- Cutadapt
- Trimmomatic ★
- fastp
- Trim Galore

They have a varying range of clipping and trimming features, but for the most part they all work similarly.

Trimming is *not always* required

There are some aligners that are available which will “soft-clip” low-quality bases or adapter sequences *during* alignment. If you are working with shorter reads (<50 bp), trimming before aligning can actually prevent the aligner from discarding poor-quality reads.

We will compare some aligners next week

Loading Trimmomatic


```
module load gcc/13.3.0-xp3epyt
```

Search for the trimmomatic module with:

```
module avail
```

Use the following to check that the program was loaded

```
module list
```

Trimmomatic options

Trimmomatic has a variety of options to trim your reads. If we run the following command, we can see some of our options.

```
trimmomatic
```

Need to specify if reads are PE or SE

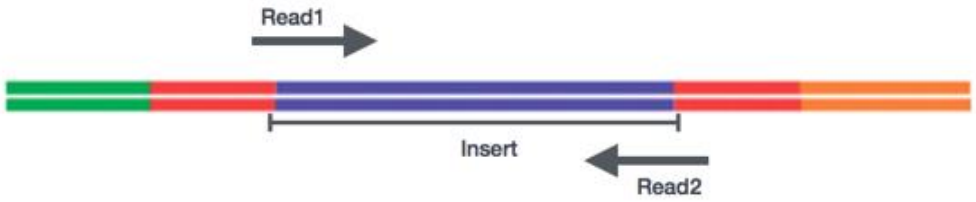
```
Usage:
      PE [-version] [-threads <threads>] [-phred33|-phr
       [-trimlog <trimLogFile>] [-summary <statsSummaryFi
       [-quiet] [-validatePairs] [-basein <inputBase> |
      <inputFile2>] [-baseout <outputBase> | <outputFile
      <outputFile1U> <outputFile2P> <outputFile2U>] <tri
  or:
      SE [-version] [-threads <threads>] [-phred33|-phr
       [-trimlog <trimLogFile>] [-summary <statsSummaryFi
       [-quiet] <inputFile> <outputFile> <trimmer1>...
```

option	meaning
inputFile1	Input reads to be trimmed. Typically the file name will contain an _1 or _R1 in the name.
inputFile2	Input reads to be trimmed. Typically the file name will contain an _2 or _R2 in the name.

Trimmomatic uses positional arguments

In PE mode expects (2) files

In SE mode expects (1) file



Other Positional arguments

In Next-Generation Sequencing (NGS), "surviving pairs" refers to the paired-end reads that successfully pass quality checks after data processing (like trimming) and can be used for further analysis, while "orphan reads" are single reads from a pair where the other read did not pass quality standards and is therefore discarded, leaving the remaining read "orphaned.".

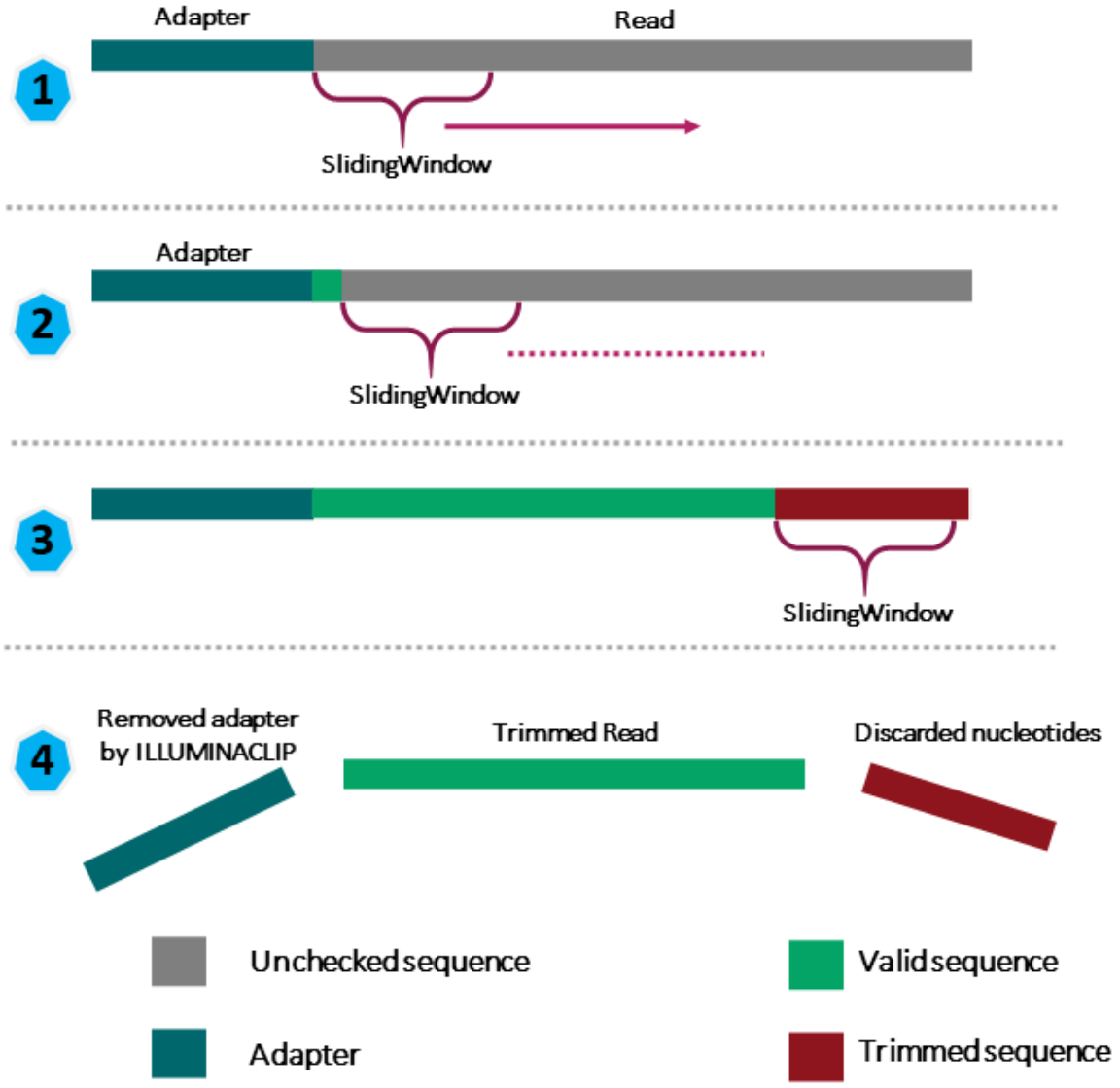
option	meaning
outputFile1P	Output file that contains surviving pairs from the _1 file.
outputFile1U	Output file that contains orphaned reads from the _1 file.
outputFile2P	Output file that contains surviving pairs from the _2 file.
outputFile2U	Output file that contains orphaned reads from the _2 file.

Trimmomatic uses the following positional arguments

In addition, trimmomatic expects to see trimming parameters:

step	meaning
ILLUMINACLIP	Perform adapter removal.
SLIDINGWINDOW	Perform sliding window trimming, cutting once the average quality within the window falls below a threshold.
LEADING	Cut bases off the start of a read, if below a threshold quality.
TRAILING	Cut bases off the end of a read, if below a threshold quality.
MINLEN	Drop an entire read if it is below a specified length.
TOPHRED33	Convert quality scores to Phred-33.

SLIDINGWINDOW



- We will use only a few of these options and trimming steps in our analysis.
- It is important to understand the steps you are using to clean your data.
- For more information about the Trimmomatic arguments and options, see the Trimmomatic manual.

http://www.usadellab.org/cms/?page=trimmomatic

A Completed Command for Trimmomatic

```
trimmomatic PE SRR_1056_1.fastq SRR_1056_2.fastq \
          SRR_1056_1.trimmed.fastq SRR_1056_1un.trimmed.fasta \
          SRR_1056_2.trimmed.fastq SRR_1056_2un.trimmed.fasta \
          SLIDINGWINDOW:4:20 ILLUMINACLIP:SRR_adapters.fa \
```

SE or PE

```
trimmomatic PE SRR_1056_1.fastq SRR_1056_2.fastq \
          SRR_1056_1.trimmed.fastq SRR_1056_1un.trimmed.fasta \
          SRR_1056_2.trimmed.fastq SRR_1056_2un.trimmed.fasta \
          SLIDINGWINDOW:4:20 ILLUMINACLIP:SRR_adapters.fa \
```

It will be taking a paired end file as input

Input file 1

Input file 2

```
trimmomatic PE SRR_1056_1.fastq SRR_1056_2.fastq \
          SRR_1056_1.trimmed.fastq SRR_1056_1un.trimmed.fasta \
          SRR_1056_2.trimmed.fastq SRR_1056_2un.trimmed.fasta \
          SLIDINGWINDOW:4:20 ILLUMINACLIP:SRR_adapters.fa \
```

Scripting etiquette

Tab on second line

\

```
trimmomatic PE SRR_1056_1.fastq SRR_1056_2.fastq \
        SRR_1056_1.trimmed.fastq SRR_1056_1un.trimmed.fasta \
        SRR_1056_2.trimmed.fastq SRR_1056_2un.trimmed.fasta \
        SLIDINGWINDOW:4:20 ILLUMINACLIP:SRR_adapters.fa \
```

\ and tab are used to make code more legible

_1.trimmed.fastq

```
trimmomatic PE SRR_1056_1.fastq SRR_1056_2.fastq \
        SRR_1056_1.trimmed.fastq SRR_1056_1un.trimmed.fasta \
        SRR_1056_2.trimmed.fastq SRR_1056_2un.trimmed.fasta \
        SLIDINGWINDOW:4:20 ILLUMINACLIP:SRR_adapters.fa \
```

- The output file for surviving pairs from the _1 file
- It will create a new file for this

_1un.trimmed.fastq

```
trimmomatic PE SRR_1056_1.fastq SRR_1056_2.fastq \
        SRR_1056_1.trimmed.fastq SRR_1056_1un.trimmed.fasta \
        SRR_1056_2.trimmed.fastq SRR_1056_2un.trimmed.fasta \
        SLIDINGWINDOW:4:20 ILLUMINACLIP:SRR_adapters.fa \
```

- The output file for orphaned reads from the _1 file
- It will create a new file for this

_2.trimmed.fastq

```
trimmomatic PE                                  1056_2.fastq \
          SRR_1056_                 rimmed.fastq SRR_1056_1un.trimmed.fasta \
          SRR_1056_2.trimmed.fastq SRR_1056_2un.trimmed.fasta \
          SLIDINGWINDOW:4:20 ILLUMINACLIP:SRR_adapters.fa \
```

- The output file for surviving pairs from the _2 file
- It will create a new file for this

_2un.trimmed.fastq

```
trimmomatic PE SRR_1056_1.fastq SRR_1
          SRR_1056_1.trimmed.fastq SRR_1056_1un.trimmed.fasta \
          SRR_1056_2.trimmed.fastq SRR_1056_2un.trimmed.fasta \
          SLIDINGWINDOW:4:20 ILLUMINACLIP:SRR_adapters.fa \
```

- The output file for orphaned reads from the _2 file
- It will create a new file for this

```
trimmomatic                                  R_1056_2.fastq \
          SR                                 SRR_1056_1un.trimmed.fasta \
          SRR_105_1.trimmed.fastq SRR_1056_2un.trimmed.fasta \
          SLIDINGWINDOW:4:20 ILLUMINACLIP:SRR_adapters.fa \
```

SLIDING WINDOW

Use a sliding window of size 4(bps) that will remove bases if their phred score is below 20

Note this is specified with :

```
trimmomatic PE SRR_1056_1.fastq SRR_
          SRR_1056_1.trimmed.fastq SR                                  a \
          SRR_1056_2.trimmed.fastq SRR_1056_un.trimmed.fasta \
          SLIDINGWINDOW:4:20 ILLUMINACLIP:SRR_adapters.fa \
```

Illuminaclip

to clip the Illumina adapters from the FASTQ files using the adapter sequences listed in SRR_adapters.fa

Note this is specified with :

Running Trimmomatic

Class Exercise

This Exercise will take ~20 mins. Please work with your neighbor if you have questions. I will begin answering questions at the 5 minute mark.

```
Input Read Pairs: 1107090
Both Surviving: 885220 (79.96%)
Forward Only Surviving: 216472 (19.55%)
Reverse Only Surviving: 2850 (0.26%)
Dropped: 2548 (0.23%)
TrimmomaticPE: Completed successfully
```

Inside of trimmomatic_exercise

- trim.sh
- SRR2589044_2.fastq.gz
- SRR2589044_1.fastq.gz
- trimmomatic_adapters

Inside of trimmomatic_adapters

- ..
- TruSeq3-SE.fa
- TruSeq3-PE.fa
- TruSeq3-PE-2.fa
- TruSeq2-SE.fa
- TruSeq2-PE.fa
- NexteraPE-PE.fa

Running Trimmomatic

Class Exercise

This Exercise will take ~20 mins. Please work with your neighbor if you have questions. I will begin answering questions at the 5 minute mark.

```
Input Read Pairs: 1107090
Both Surviving: 885220 (79.96%)
Forward Only Surviving: 216472 (19.55%)
Reverse Only Surviving: 2850 (0.26%)
Dropped: 2548 (0.23%)
TrimmomaticPE: Completed successfully
```

Phil Ewels

Senior Product Manager for Open Source Software at Seqera

Phil Ewels is Product Manager for Open Source at Seqera. He holds a PhD in Molecular Biology from the University of Cambridge, UK. Before joining Seqera in 2022, Phil worked at the National Genomics Infrastructure (NGI) at SciLifeLab in Stockholm, Sweden. He and his team focussed on developing and scaling up new lab protocols and bioinformatics solutions. This involved developing new analysis pipelines that could scale to very high volumes whilst adhering to high standards of reproducibility and reusability. It was through this work that Phil became involved in the Nextflow project and eventually co-founded the nf-core community. Phil's career has spanned many disciplines from lab work and bioinformatics research in epigenetics, through to software development and community engagement. He is passionate about open-source software and has a soft spot for tools with a focus on user-friendliness. He is the author of MultiQC, SRA-Explorer, QCFail.com, and several other pet projects.

SRA Explorer

https://sra-explorer.info/

Mini web application to explore the NCBI Sequence Read Archive and easily access downloads for data, either as .sra files from the NCBI or as .fastq via the EBI ENA.

MULTIQC

MultiQC is a tool to create a single report with interactive plots for multiple bioinformatics analyses across many samples.

https://seqera.io/examples/rna-seq/multiqc_report.html

Short Exercise

Module: py-multiqc/1.15-fmpaaj7

File location:
/gpfs1/cl/mmg3320/course_materials/multiqc_example

Run the following command after navigating inside:

multiqc -n test_multiqc .