

fastqc_parser

Purpose:

fastqc_parser can parse the output of FASTQC reports: consolidate the results and extract the relevant information from all the output files. Depending on input arguments, fastqc_parser can either run fastqc on raw fastq files and then parse the output. Alternatively, it can be run directly on fastqc output files. To run this script, the following dependencies are required: PYTHON, pandas module, and FASTQC.

Usage:

fastqc_parser.py <run fastqc> [--fastqc_folder] [--param] [-- input_path] [-- output_path] [-- files]

Required arguments are indicated within <> and one of the allowed options has to be entered compulsorily. Optional arguments are indicated within square brackets. If the optional arguments are not specified by a user, then the default values will be used. Detailed information regarding required and optional arguments are indicated below.

Required arguments	
Argument	Allowed Options
--run_fastqc	1) Yes - to run fastqc and then parse the output. 2) No – when fastqc reports are done already, the relative path to the folder containing them can be specified.
--fastqc_folder	If the option of --run_fastqc is “No”, then this is the folder containing all fastqc reports.

Required arguments	
Argument	Default Option
--input_path	Current working directory. If the FASTQ files are not in the current working directory, then the path to input FASTQ files can be specified here.
--output_path	Current working directory. The output from fastqc_parser is stored in the current directory under folder name “output”. Users can also specify a different folder path.
--param	FAIL (Allowed options: “PASS”, “WARN”, “FAIL”)
--files	None. If the files are located in different locations, the absolute path of the files can be specified. Note: if --input_path is specified, --files cannot be specified. If --files argument is specified, files in the current working directory will not be analyzed, even though the output directory will be the current working directory.