

Preparing Job File

You need to provide two JSON files that describes your job:

`${sample-name}.inputs.json`

```
{
  "Velopipe2.sampleName": "RU263",
  "Velopipe2.countMatrix": "s3://dp-lab-data/collaborators/pi/project/sample/..._dense.csv",
  "Velopipe2.bam": "s3://dp-lab-data/collaborators/pi/project/sample/..._Aligned.out.sorted.bam",
  "Velopipe2.numOfChunks": 100,
  "Velopipe2.gtf": "s3://seqc-public/genomes/mm38_long_polya/annotations.gtf",
  "Velopipe2.barcodeWhitelist": "s3://seqc-public/barcodes/ten_x_v3/flat/3M-february-2018.txt",
  "Velopipe2.cbCorrection": "s3://dp-lab-test/seqc/joe/Ru263/seqc-results-v2/941_Ru263_IGO_09507_10_cb-correction.csv.gz",
  "Velopipe2.umiCorrection": "s3://dp-lab-test/seqc/joe/Ru263/seqc-results-v2/941_Ru263_IGO_09507_10_umi-correction.csv.gz"
}
```

- `Velopipe2.countMatrix`: a file from which filtered barcodes will be extracted
 - a dense matrix file (e.g. `*_dense.csv`) or
 - a barcode file that is used to construct the sparse matrix (e.g. `*_sparse_counts_barcodes.csv`)
- `Velopipe2.bam`: BAM file generated by SEQC
- `Velopipe2.numOfChunks`: BAM file will be split into multiple chunks to parallelize.
- `Velopipe2.gtf`: GTF file
 - Human: `s3://seqc-public/genomes/hg38_long_polya/annotations.gtf`
 - Mouse: `s3://seqc-public/genomes/mm38_long_polya/annotations.gtf`
- `Velopipe2.barcodeWhitelist`: a barcode whitelist
 - 10x v2: `s3://seqc-public/barcodes/ten_x_v2/flat/737K-august-2016.txt`
 - 10x v3: `s3://seqc-public/barcodes/ten_x_v3/flat/3M-february-2018.txt`
- `Velopipe2.cbCorrection`: CB correction file generated by SEQC
- `Velopipe2.umiCorrection`: UMI correction file generated by SEQC

Note that SEQC produces a read-name sorted BAM file (this is different from position sorted).

`${sample-name}.labels.json`

```
{
  "pipelineType": "Velopipe2",
  "project": "Project 193",
  "sample": "1469_TGFb_LCC-TRL_1_P193",
  "owner": "chunj",
  "destination": "s3://dp-lab-data/Siting/TGFb_LCC-TRL1",
  "transfer": "-",
  "comment": "RNA Velocity"
}
```

- `project`: project ID retrieved from SCRI database
- `sample`: sample name
- `destination`: AWS S3 location where the final output files (e.g. loom) should be saved