Managing Data & Projects for Genomics

Quick Reference · Keep these practices in every project

MDIBL CGDS Core
CGDS@mdibl.org

FAIR DATA PRINCIPLES

F — Findable

Consistent names · logical folders · searchable metadata · persistent IDs (DOI, GEO, Ensembl)

A — Accessible

Documented paths · readable formats · open without special software · clear metadata

I — Interoperable

Open formats (`.csv` , `.tsv`) · standard gene IDs · numeric data as numbers, not PDFs

R — Reusable

README · samplesheet · tidy structure · descriptive headers · documented pipeline + parameters

Gene IDs: Use Ensembl IDs (`ENSG00000141510`) as primary keys — never gene symbols. Symbols change across annotation versions and cause silent mismatches.

METADATA — WHAT TO CREATE

DOCUMENT	CONTENTS	LOCATION
`samplesheet.csv`	One row per file: filename, condition, treatment, replicate, collection date	`data/`
`design.tsv`	One row per sample: condition + treatment variables for analysis code	`analysis/input/`
`README.txt`	Project overview, directory map, file naming convention, tool versions	Project root

SAMPLESHEET STRUCTURE (TIDY DATA)

FILENAME	LIGHT	GENOTYPE	MEDIA	TIMEPOINT	DATE
LD_phyA_off_t04_2020-08-12.norm.txt	LD	phyA	off	t04	2020-08-12
LD_phyA_on_t04_2020-07-14.norm.txt	LD	phyA	on	t04	2020-07-14
SD_phyB_off_t04_2020-08-13.norm.txt	SD	phyB	off	t04	2020-08-13
· · · one row per file · · ·					

One variable per column · one observation per row · one value per cell

README MUST-HAVES

- ☐ Project title and one-sentence description
- ☐ Organism (species · strain/cell line)
- ☐ Reference genome build + annotation version
- ☐ Date created and point of contact (name + email)
- ☐ Directory map — what each folder contains
- ☐ File naming convention — every field defined
- ☐ Samplesheet location and what columns mean
- ☐ Pipeline name + version + key parameters
- ☐ Statistical cutoffs and normalization method
- ☐ Known issues or data caveats

FILE NAMING RULES

Pattern: `{light}_{genotype}_{media}_{timepoint}_{date}.{datatype}.txt`

RULE	✓ DO THIS	✕ NOT THIS
No spaces	`Mock_DMSO_rep1.fastq.gz`	`Mock DMSO rep1.fastq.gz`
No special characters	`results_v2.csv`	`results (v2)!.csv`
Zero-pad	`s01, s02, s12`	`s1, s2, s12`
ISO 8601 date	`2024-07-16`	`07-16-24 · JUL-16`
Consistent case	`LD_phyA · LD_phyB`	`LD_phyA · ld_phyb`
Content-based	`PCA_all_samples.png`	`fig_3_a.png`
Variables first	`LD_phyA_on_t04_2024-07.norm.txt`	`2024-07-14_s1_LD.txt`

PROJECT DIRECTORY TEMPLATE

```
project_name/
├── data/           ← raw data + metadata (READ-ONLY)
│   ├── raw/        ← unprocessed files, never modify
│   ├── norm/       ← normalized / processed outputs
│   ├── samplesheet.csv
│   └── *.fastq.gz / reference files
├── analysis/       ← code, pipeline inputs & outputs
│   ├── script.Rmd
│   ├── input/
│   └── output/
├── doc/            ← protocols, notes, documentation
├── results/        ← final figures and tables for paper
└── README.txt      ← who/what/when/where/how + map
```

- One project per directory
- Separate raw from processed — treat raw as read-only
- No `OLD/` , `misc/` , or version number suffixes
- Name files by content, not manuscript position

ONE THING TO FIX THIS WEEK

No README?
Write one for your most-used project. Start with: a description, where is the raw data, what does each folder contain.

No samplesheet?
Create a `.csv` with one row per file: sample name · condition · replicate · file name. Move collection dates out of filenames and into this table.

Inconsistent file names?
Write down the convention you wish you had used — pick one and document it in your README. Apply it to all new files going forward.

Results in .xlsx?
Export your key analysis result to `.csv`

Gene symbols as IDs?
Add an `ensembl_id` column (`ENSG…`)

No folder structure?
Start by separating datatypes into `raw/` and `results/` .