

DATA MANAGEMENT PLAN

План управління даними: DMP

ЯК НЕ ЗАГУБИТИСЯ У ВЛАСНИХ ДАНИХ: ФАЙЛИ, ПАПКИ ТА ВЕРСІЇ

Добре організовані дослідницькі дані починаються не з репозитарію і не з моменту публікації. Вони починаються з того, як ви називаєте файли, будujete папки та відрізняєте одну версію від іншої.

У невеликому проекті хаотична структура може здаватися несерйозною проблемою. Але коли з'являються кілька дослідників, десятки файлів, декілька раундів збору даних, різні версії таблиць і скриптів, дуже швидко виникають типові питання:

- який файл є актуальним;
- що було змінено після попереднього аналізу;
- де зберігаються первинні дані;
- яку таблицю використано для конкретного рисунка у статті;
- чи можна відтворити результати через рік;
- чи зрозуміє структуру проекту інший член команди.

Тому організація файлів — це частина відтворюваності та належного управління даними, а не просто питання зручності.

Побудуйте зрозумілу структуру папок

Структура папок повинна відображати логіку роботи над дослідженням. Не існує єдиної правильної схеми, але бажано, щоб вона була сталою протягом усього проекту.

Наприклад:

```
Project_Name/  
├── 01_Admin/  
├── 02_Protocol/  
├── 03_Raw_Data/  
├── 04_Processed_Data/  
├── 05_Analysis/  
├── 06_Code/  
├── 07_Figures_Tables/  
├── 08_Publications/  
├── 09_Documentation/  
└── 10_Data_for_Sharing/
```

Нумерація допомагає зберігати однаковий порядок папок у різних операційних системах.

Для невеликого проекту структура може бути значно простішою. Важливий не масштаб, а принцип: первинні дані, оброблені дані, аналіз і фінальні результати не повинні змішуватися в одному місці.

Не перезаписуйте первинні дані

Папка з первинними даними (*Raw Data*) повинна містити дані максимально близькі до того вигляду, у якому вони були отримані.

Наприклад:

- вихідні файли вимірювального обладнання;
- первинний експорт анкети;
- оригінальні фотографії або мікрофотографії;
- аудіозаписи інтерв'ю;
- первинні транскрипти;
- отримані від партнерів набори даних.

Після початку аналізу ці файли бажано не редагувати і не перезаписувати.

Очищення, анонімізацію, перекодування, нормалізацію або інші зміни краще виконувати на копіях і зберігати результат окремо як *Processed Data*.

Так ви завжди можете повернутися до вихідного матеріалу і перевірити, як саме були отримані результати.

Давайте файлам назви, які щось означають

Назва файлу повинна дозволяти зрозуміти його зміст без необхідності відкривати файл.

Погано:

```
data.xlsx
new_data.xlsx
table_new.xlsx
final.xlsx
final2.xlsx
final_revised_last.xlsx
```

Краще:

```
TRACE_SurveyResponses_2026-09-15_raw.csv
TRACE_SurveyResponses_2026-09-20_clean_v01.csv
Ga2O3_XRD_SampleA_2026-10-03_raw.xy
Ga2O3_XRD_SampleA_processed_v02.csv
InterviewCoding_2026-09-28_v03.xlsx
```

Корисна загальна схема:

[Проект]_[Тип даних]_[Дата]_[Статус/версія]

Наприклад:

RDM2026_InterviewData_2026-10-06_v01.docx

Не кожен елемент обов'язковий. Головне — щоб команда використовувала однакову систему протягом усього проєкту.

Дату краще записувати однаково

Для назв файлів зручно використовувати формат:

YYYY-MM-DD

Наприклад:

2026-10-06

а не:

06.10.26
6October
Oct6

Такий формат має ще одну перевагу: файли автоматично сортуються у хронологічному порядку.

Не використовуйте «final_final»

Слова *final*, *new*, *latest*, *corrected* швидко втрачають сенс.

Якщо файл змінюється, краще використовувати номер версії:

Dataset_v01.csv
Dataset_v02.csv
Dataset_v03.csv

Для суттєво нової версії можна застосувати:

Dataset_v1.0.csv
Dataset_v1.1.csv
Dataset_v2.0.csv

Наприклад:

- **v0.1** — робоча версія;
- **v1.0** — перша завершена версія;
- **v1.1** — незначні виправлення;
- **v2.0** — суттєво змінений набір.

Для великих проєктів або роботи з кодом краще використовувати систему контролю версій, наприклад Git.

Фіксуйте, що саме було змінено

Номер версії корисний лише тоді, коли зрозуміло, що відрізняє одну версію від іншої.

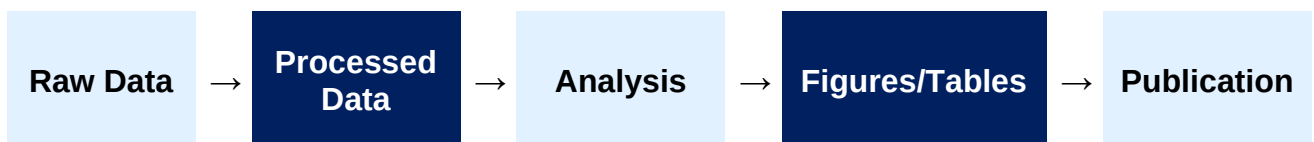
Для важливих наборів можна вести простий **change log**:

Версія	Дата	Зміни	Відповідальний
v01	2026-09-15	Первинний експорт	І. Петренко
v02	2026-09-18	Видалено дублікати	І. Петренко
v03	2026-09-20	Узагальнено вік, проведено анонімізацію	О. Іваненко

Цю інформацію можна зберігати у README-файлі.

Розділяйте дані, аналіз і результати

Корисно підтримувати логічний зв'язок:



Наприклад, якщо у статті є рисунок Figure_3.png, бажано мати можливість зрозуміти:

1. з яких даних його побудовано;
2. яка версія даних використовувалася;
3. яким скриптом або програмою проведено аналіз;
4. які параметри були застосовані.

Це суттєво спрощує повторну перевірку результатів, підготовку відповіді рецензентам і повторне використання даних.

Якщо над проектом працює команда

Домовтеся про правила на початку, а не після появи сотні файлів.

Визначте:

- спільну структуру папок;
- систему назв;
- формат дат;
- правила версій;
- хто може змінювати первинні дані;
- де знаходиться актуальна робоча версія;
- хто відповідає за резервне копіювання;
- хто готує фінальний набір до архівування.

Добре, якщо ці правила коротко описані у README або DMP.

Не забувайте про чутливі дані

Структура папок також повинна враховувати рівень доступу.

Наприклад:

```
Research_Project/  
├── Raw_Data_RESTRICTED/  
├── Consent_Forms_RESTRICTED/  
├── Pseudonymisation_Key_RESTRICTED/  
├── Processed_Data/  
└── Data_for_Sharing/
```

Форми інформованої згоди, контактні дані учасників та ключі псевдонімізації не повинні зберігатися разом із набором, підготовленим для поширення.

Коротка перевірка

Перед тим як продовжити роботу з проектом, запитайте себе:

- ☐ Чи можу я за 30 секунд знайти первинні дані?
- ☐ Чи зрозуміло, яка версія файлу є актуальною?
- ☐ Чи збережені первинні дані без змін?
- ☐ Чи зрозумілі назви файлів без їх відкривання?
- ☐ Чи можна відтворити шлях від первинних даних до рисунка або таблиці у статті?

- ☐ Чи зрозуміє структуру папок новий член команди?
- ☐ Чи відокремлені конфіденційні дані від матеріалів, які можна поширювати?

Якщо на більшість цих запитань відповідь «так» — ваші файли вже керуються значно краще, ніж здається.

Корисні ресурси:

1. Інструкція зі зберігання дослідницьких даних Бердянського державного педагогічного університету. Zenodo. <https://doi.org/10.5281/zenodo.18210810>
2. Шаблон журналу доступу до дослідницьких даних Бердянського державного педагогічного університету. Zenodo. <https://doi.org/10.5281/zenodo.18210845>
3. [OpenAIRE: організація файлів і версій](#)

ЗБЕРІГАННЯ, РЕЗЕРВНЕ КОПІЮВАННЯ ТА БЕЗПЕКА

Зберігання даних і резервне копіювання — це різні, хоча й пов'язані процеси. Зберігання означає робоче середовище, у якому дані використовуються під час дослідження, тоді як резервна копія потрібна для відновлення інформації у випадку втрати, пошкодження, помилкового видалення або технічного збою. Тому важливі дослідницькі дані не повинні існувати лише в одному місці — наприклад, тільки на ноутбучі, флешці чи зовнішньому диску.

Для робочих даних бажано використовувати захищені інституційні або інші надійні електронні середовища з контрольованим доступом. Резервні копії слід зберігати окремо від основної копії та регулярно оновлювати. Для критично важливих даних корисно дотримуватися логіки кількох незалежних копій у різних середовищах.

Особливі вимоги застосовуються до персональних, конфіденційних і чутливих даних. Для них необхідно:

- обмежувати доступ лише тими особами, яким він справді потрібний;
- використовувати надійні паролі та двофакторну автентифікацію;
- за потреби застосовувати шифрування;
- зберігати форми інформованої згоди та ключі псевдонімізації окремо від основного набору даних;
- не використовувати відкриті або неконтрольовані сервіси для зберігання чутливої інформації без оцінки ризиків.

Перед початком дослідження варто визначити: де зберігатимуться робочі дані, де буде резервна копія, хто матиме доступ і як довго дані потрібно зберігати після завершення проєкту.

Корисні ресурси:

1. Інструкція зі зберігання дослідницьких даних Бердянського державного педагогічного університету. Zenodo. <https://doi.org/10.5281/zenodo.18210810>
2. [OpenAIRE Research Data Management Guides](#)
3. Пам'ятка для дослідників Бердянського державного педагогічного університету щодо використання стороннього контенту. Zenodo. <https://doi.org/10.5281/zenodo.18210886>
4. Протокол обмеженого доступу до дослідницьких даних Бердянського державного педагогічного університету. Zenodo. <https://doi.org/10.5281/zenodo.18210919>
5. Рекомендацій з оцінки ризиків при розкритті дослідницьких даних Бердянського державного педагогічного університету. Zenodo. <https://doi.org/10.5281/zenodo.18211044>
6. Інструкція щодо класифікації дослідницьких даних за рівнем доступу. Zenodo. <https://doi.org/10.5281/zenodo.23164251>
7. Інструкція щодо оцінки можливості розкриття дослідницьких даних. Zenodo. <https://doi.org/10.5281/zenodo.23164288>
8. Рекомендації щодо електронного зберігання дослідницьких даних у БДПУ. Zenodo. <https://doi.org/10.5281/zenodo.23164346>
9. Політика анонімізації та псевдонімізації дослідницьких даних у БДПУ. Zenodo. <https://doi.org/10.5281/zenodo.23165010>
10. Інструкція щодо проведення анонімних та конфіденційних опитувань студентів у БДПУ. Zenodo. <https://doi.org/10.5281/zenodo.23165034>
11. Протокол підготовки дослідницьких даних до відкритого поширення. Zenodo. <https://doi.org/10.5281/zenodo.23165058>
12. Чекліст оцінки ризику повторної ідентифікації дослідницьких даних перед поширенням. Zenodo. <https://doi.org/10.5281/zenodo.23165082>
13. Форма самооцінки етичних ризиків наукового дослідження. Zenodo. <https://doi.org/10.5281/zenodo.23165424>

README, Data Dictionary і метадані

Файл може зберегтися через десять років. Але чи буде зрозуміло, що в ньому, звідки взялися ці дані і як ними користуватися?

Саме для цього потрібна документація до даних. Вона допомагає не лише іншим дослідникам, а й самому автору через кілька місяців або років відновити логіку роботи з набором. Добре задокументовані дані легше перевіряти, повторно використовувати, передавати колегам і готувати до публікації в репозитарії.

Найпростіший і найкорисніший документ — **README-файл**. Він зазвичай пояснює, що містить набір даних, як вони були отримані, які файли входять до набору, як вони організовані, яке програмне забезпечення потрібне для роботи з ними, які трансформації були виконані, які існують обмеження доступу та як правильно цитувати набір.

README може містити, наприклад:

- назву та короткий опис набору;
- авторів і контактну інформацію;
- структуру папок і файлів;
- методи збору та обробки даних;
- програмне забезпечення або обладнання, що використовувалося;
- інформацію про версії;
- умови доступу й ліцензію;
- рекомендоване цитування;
- відомі обмеження або особливості даних.

Якщо набір містить багато змінних, категорій або кодів, корисно додати окремий **Data Dictionary** — словник даних. Він пояснює, що означає кожна змінна, у яких одиницях вона вимірюється, які значення допустимі, як закодовані категорії та що означають пропущені значення.

Наприклад:

```
age_group — вікова група учасника;  
1 = 18–24, 2 = 25–34, 3 = 35–44;  
NA = дані відсутні.
```

Для експериментальних даних словник може містити назви параметрів, одиниці вимірювання, точність приладу, умови експерименту та скорочення. Для якісних даних — коди тематичного аналізу, принципи транскрибування, правила позначення пауз або анонімізації.

Важливо також пам'ятати про **метадані** — структуровану інформацію про сам набір даних: хто його створив, коли, у межах якого проекту, якими методами, з якими умовами доступу та за якою ліцензією. Саме метадані допомагають іншим знайти набір, зрозуміти його зміст і правильно інтерпретувати.

Практичне правило просте: людина, яка не брала участі у вашому проекті, повинна мати змогу зрозуміти структуру й зміст даних без додаткових усних пояснень.

Корисні ресурси:

1. [Cornell: Writing READMEs for Research Data](#) — практичні рекомендації щодо структури README-файлів для дослідницьких даних.
2. [FAIRsharing](#) — каталог стандартів, репозитаріїв і політик, який допомагає знайти відповідні стандарти метаданих і опису даних для конкретної дисципліни.

Cornell підкреслює, що README допомагає забезпечити правильну інтерпретацію даних у майбутньому як самим автором, так і іншими користувачами. FAIRsharing, своєю чергою, корисний тоді, коли потрібно зрозуміти, які саме стандарти метаданих, формати чи словники прийняті у вашій галузі.