Datan organisoinnin ABC

Siiri Fuchs, Hanna Koivula, Tuija Korhonen, Tanja Lindholm, Päivi Rauste, Liisa Siipilehto

Sisältö

Osa 1

1. Kansiorakenne
2. Nimeäminen
3. Versionhallinta
4. Read me
5. Pidä kirjaa, siitä mitä teet
6. Tehtävä 1

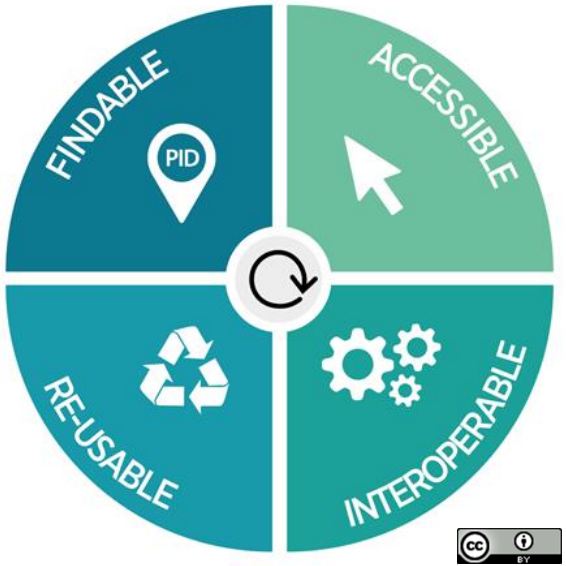
Osa 2

1. Miten pitää taulukkomuotoinen tiedosto siistinä?
2. Muotoilun, kommenttien ja yksiköiden käyttäminen
3. Muuttujien nimeäminen
4. Nolla-arvojen & puuttuvien tietojen merkitseminen
5. Metatiedon lisääminen taulukkoon
6. Tehtävä 2

Hyvin organisoitu ja dokumentoitu data luo perustan koko tutkimuksen elinkaarelle.

Kun data on hyvin organisoitu ja dokumentoitu, sitä on helppoa jakaa, avata ja käyttää uudelleen.

Noudattamalla muutamaa perusasiaa datan käsittelyssä ja dokumentoinnissa, olet jo monta askelta lähempänä FAIR-dataa.



HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Miksi organisoida ja dokumentoida?

1. Hyvin laaditut tiedostojen nimet ja kansiorakenteet helpottavat datatiedostojen löytämistä ja seuraamista
2. Dataa on helpompi käyttää, jakaa, avata ja arkistoida
3. Standardoidut käytännöt lisäävät aineiston arvoa ja jatkokäytön mahdollisuuksia
4. Hyvä dokumentaatio vähentää datan väärän tulkinnan riskiä
5. Huolelliseen dokumentaatioon käytetty aika säästää aikaa sekä projektin aikana että julkaisuvaiheessa
6. Hyvä metadata lisää datan löydettävyyttä

Hyvin organisoitu ja hyvin dokumentoitu

Hyvässä dokumentaatiossa on otettu huomioon seuraavat asiat:

1. Aineistonkeruun menetelmistä: näytteenotto, miten data on kerätty, mitä laitteita ja ohjelmistoja on käytetty
2. Laadunvarmistusmenetelmät
3. Kansiorakenne
4. Versionhallinta
5. Tiedot pääsy- ja käyttöehdoista tai aineistojen luottamuksellisuudesta
6. Muuttujien, tietueiden ja niiden arvojen nimet, merkinnät ja kuvaukset
7. Selitys tai määritelmä käytetyistä koodistoista ja luokitusjärjestelmistä
8. Määritelmät käytetyistä erikoistermeistä tai lyhenteistä
9. Puuttuvien arvojen koodit ja syyt niihin

Kansiorakenteen suunnittelu

- Kenelle suunnittelet datan organisointijärjestelmää? Itsellesi, tutkimuksen tekijälle, projektin tarpeisiin, yhteistyökumppaneille?
- Onko projekti lyhyt vai pitkäkestoinen? Riittääkö yksinkertainen kansiorakenne vai tarvitaanko monitahoista rakennetta?
- Kenellä tulee olla pääsy kansioihin? Luettele kaikki henkilöt (myös potentiaaliset), joiden on päästävä käsiksi järjestelmässäsi oleviin tiedostoihin.
- Kopioiden ja varmuuskopioiden ottaminen sekä pääsynhallinta on tehokasta organisoidun kansiorakenteen avulla (esim. jos projektissa käsitellään sensitiivistä dataa, tietty kansio voidaan suojata salasanalla).

Toimiva kansio- ja tiedostorakenne

- Tee kullekin projektille oma kansio nimen ja ajankohdan mukaan (esim. lyhenne + vuosi)
- Datatiedostojen lisäksi tee erilliset kansiot projektin hallinnointiin, menetelmille, tekstitiedostoille jne.
- Mieti kenellä tulee olla pääsy kansioon
- Käytä yksilöllisiä tiedostojen ja kansioiden nimiä.
  - Tiedostojen nimet eri kansioiden alla eivät saa mennä sekaisin, jos kansiorakenne puretaan (älä siis käytä samoja tiedostojen nimiä eri kansioissa ja toisaalta nimeä tiedostot niin, että tiedät, mihin kansioon ne kuuluvat (joku tunniste).
- Oikea tasapaino matalan ja syvän kansiohierarkian välillä auttaa löytämään oikean tiedoston.
  - Vältä kansioita kansioiden sisällä. Älä esimerkiksi tee eri vuosille eri kansioita siitä huolimatta että haluat pitää vuodet erillään. Sen sijaan nimeä tiedostot yksilöllisesti ja pidä ne samassa kansiossa.

Hyvä kansiorakenne & sisältö

Hyvä rakenne pitää sisällään vähintään seuraavat elementit:

1. Ainutlaatuinen (unique) pääkansio projektille
2. Koodit
3. Data
4. Readme-dokumentti, joka sisältää kaiken tärkeän informaation projektista (näitä voi olla useita). Kansion ylätasolla pitää olla vähintään yksi readme, josta käy ilmi vähintään hallinnolliset asiat

https://mitcommlab.mit.edu/be/commkit/file-structure/#ChooseScaleAim

Data voidaan jakaa eri kansioihin esimerkiksi näin:

- Yksilöllinen pääkansio projektille
  - Koodi
  - Data
    - Raakadata
    - Muokattu data
    - Lopullinen data
- Readme

https://github.com/mitcommlab/Coding-Documentation/blob/master/File-Structure-Case-Studies.md#case-study-2-a-simple-hierarchy

Kansiorakenne-esimerkkejä


```
project/
  code/                code needed to go from input files to final results
  data/                raw and primary data (never edit!)
     raw_external/
     raw_internal/
     meta/
  doc/                 documentation of the study
  intermediate/        output files from intermediate analysis steps
  logs/                logs from the different analysis steps
  notebooks/           notebooks that document your day-to-day work
  results/             output from workflows and analyses
     figures/
     reports/
     tables/
  scratch/             temporary files that can safely be deleted or
  README.txt           file and folder description
```

Lähde: RDMKit, Elixir.

Case Study 2: A Simple Hierarchy

```
PROJECT/
├── bin/            <- compiled binaries.
├── data/
│   ├── raw/
│   └── clean/
│
├── figures/        <- figures used in place of a "results" folder.
├── scripts/
│   ├── process/    <- scripts to maniuplate data between raw, cleaned, final stages.
│   └── plot/         <- intermediate plotting.
│
├── src
│   ├── model1/     <- various experimental models.
│   ├── model2/
│   └── model3/
│
├── LICENSE
├── Makefile
└── readme.md
```

Lähde: Three examples of file structures for different project types

20221202_Mountain_EXP2_Kilpis_DATA_V01.xls

Tiedostojen nimeäminen

- **Tiedoston nimi on tiedoston päätunniste**: lyhyet & kuvaavat tiedostonimet kertovat mitä tiedosto sisältää ja helpottavat datan järjestämistä, mutta vain jos nimeäminen on ollut **johdonmukaista**.
- Tiedostojen **nimeämiskäytäntö kannattaa sopia jo projektin alkuvaiheessa koko ryhmän kesken** ja nimiin kannattaa sisällyttää projektille merkitykselliset elementit. Tärkeää on, että jokainen projektissa seuraa sovittua tiedostojen nimeämismallia.

Esimerkkielementtejä sisällytettäväksi tiedostojen nimiin

- Luomispäivä
- Projektin numero / kokeen numero, akronyymi
- Datatyypit (näyte ID, analyysi, olosuhteet, muutokset ym.)
- Paikka / koordinaatit
- Tekijän nimi / nimikirjaimet
- Versionumero
- Viimeiset kolme kirjainta kertoo tiedostoformaatin (esim. .xls, .rtf, .mov, .tif, .doc)

Lähteet:
https://docs.csc.fi/data/datasets/metadata-and-documentation/#data-organization
https://rdmkit.elixir-europe.org/data_organisation#what-is-the-best-way-to-name-a-file

Vinkkejä tiedostojen nimeämiseen

20221202_Mountain_EXP2_Kilpis_DATA_V01.xls

1. Tasapainoile elementtien määrän kanssa: liian monta tekee nimestä vaikeasti ymmärrettävän, liian vähän puolestaan geneerisen.
2. Järjestä elementit yleisestä spesifiseen.
3. Käytä merkityksellisiä lyhenteitä.
4. Käytä alaviivaa (_), väliviivaa (-) tai isoa alkukirjainta elementtien erottamiseen nimessä. Älä käytä välilyöntiä tai erikoismerkkejä: ?!& , * % # ; * ( ) @$ ^ ~ ‘ { } [ ] < >.
5. Käytä päivämääräformaattina (ISO8601): YYYYMMDD (vuosi, kk, päivä), ja aika, jos tarpeen HHMMSS (tunnit, minuutit, sekunnit).
6. Sisällytä versionumero nimeen, jos tarpeen: vähintään kaksi numeroa (V02) ja pidennä, jos tarpeen pienemmille korjauksille (V02-03). Ensimmäiset nollat mahdollistavat, että tiedostot järjestyvät oikein.
7. Kirjoita nimeämiskäytäntösi ylös ja selitä lyhenteet dokumentaatiossasi (esim. Readme-tiedosto).
8. Jos joudut uudelleen nimeämään ison määrän tiedostoja hallitaksesi tiedostoja helpommin, on mahdollista käyttää sovelluksia esim. Bulk Rename Utility (Windows, free), Renamer4Mac (Mac).

ESIMERKKI: Tunturiprojekti, koe 2, tehty Kilpisjärvellä, datatiedosto luotu 2.12.2022

- Tiedoston nimi: 20221202_Mountain_EXP2_Kilpis_DATA_V01.xls
- Selitys: Time_ProjectAbbreviation_ExperimentNumber_Location_TypeOfData_VersionNumber

20221202_Mountain_EXP2_Kilpis_DATA_V01.xls

20221202_Mountain_EXP2_Kilpis_DATA_V02.xls

Versionhallinta auttaa pitämään datan järjestyksessä!

- Versionhallinta voidaan tehdä joko manuaalisesti, jolloin tiedoston nimen loppuun liitetään juokseva numero (_v03), tai automaattisesti, mikä on suositeltavin tapa.
  - Automaattinen versionhallinta voidaan toteuttaa Gitin, GitHubin tai GitLabin kaltaisilla ohjelmistoilla (organisaatiosi saattaa tarjota integroitua ratkaisua).
- Voit käyttää myös pilvitallennusratkaisuja, jotka yleensä tarjoavat automaattisen tiedostojen versioinnin. Versionhallintajärjestelmiä on useita, sekä avoimen lähdekoodin että kaupallisia, suljettuja toteutuksia.

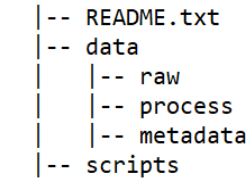
20221202_Mountain_EXP2_Kilpis_DATA_V01.xls

20221202_Mountain_EXP2_Kilpis_DATA_V02.xls

Tärkeää muistaa versionhallinnassa

- Versionhallinta tuottaa tiedostosta (muutetun) kopion, joka on merkitty yksiselitteisesti versionumerolla. Versionhallinta mahdollistaa palautuksen edelliseen versioon, mikä on tärkeää tietojen jäljitettävyyden, muokkausten seurannan ja virheiden korjaamisen kannalta.
- Kun datatiedostoista tehdään uusia versioita, on tärkeää säilyttää kopio alkuperäisestä raakadatasta.
- Dataa jaettaessa voi myös olla hyödyllistä tarjota sekä käsittelemättömät että käsitellyt versiot datastasi ja liittää mukaan joko koodi tai selitykset lopullisen version tuottamiseksi.
- Suunnittele ja sovi, mitkä versiot datasta arkistoidaan ja/tai julkaistaan.

README.txt tai LUEMINUT.txt - tiedosto



- README- tai LUEMINUT- tiedosto sitoo datakokonaisuuden osaset yhteen. Siihen voi kerryttää ja kuvata datan historiatietoa (lineage), eli erillisten tiedostojen väliset yhteydet, keruumenetelmät, datan laatutietoja, käyttötarkoitus ja rajoitukset.
- *LueMinut-tiedostoon kirjataan datan käsittelyssä (ja versioinnissa) syntyvä dokumentaatio ja datan laatuun liittyvää tietoa*
- Toimii “hiljaisen tiedon” säilyttäjänä itselle ja tutkimusryhmälle
- Helpottaa datan julkaisua ja ymmärrettävyyttä
- Sen avulla ohjeistetaan datan uudelleenkäyttöä

Lisäksi tiedoston tai tietokannan tason metadata kuvaa, mm. miten tiedostot liittyvät toisiinsa ja missä muodossa ne ovat. Kansiotason readme.txt-tiedoston avulla on mahdollista kuvata kaikki projektin tiedostot ja kansiot.

https://carpentries-incubator.github.io/spreadsheet-humanities-lesson/02-common-mistakes/

ReadMe *LUEMINUT* tiedosto

```
|-- README.txt
|-- data
|   |-- raw
|   |-- process
|   |-- metadata
|-- scripts
```

- ReadMe tiedosto on yksinkertainen tekstitiedosto, kuten .txt
- Se tulee nimetä selkeästi hyviä nimeämiskäytänteitä käyttäen (ei pakollista)
- Se sisältää tiedon milloin se on luotu tai päivitetty
- Päivämäärä on ISO 8601 muotoa: YYYYMMDD
- Tiedostosta löytyy tekijä sekä tekijän yhteystiedot

Datasettien osalta siinä on tiedot vähintään seuraavista

- Milloin data on kerätty/tuotettu
- Datan lisenssi
- Linkit julkaisuun, joka käyttää dataa
- Linkit avoimesti saataviin datasetteihin
- Viite datasettiin
- Tiedostojen kuvaukset
- Viittaus tiedostoon, jossa on esim.:
  - muuttujien tai luokitusten kuvaus (code book)
  - datassa käytetyt ulkopuoliset sanastot
  - menetelmäkuvaukset ja datan/tiedostojen yhteys niihin

Pidä kirjaa siitä mitä teet

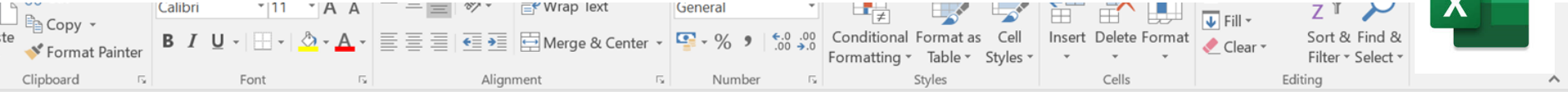
Toistettavuuden kannalta on tärkeää pitää kirjaa kaikista datan muokkausvaiheista

- Älä koskaan muokkaa alkuperäistä nk. raakadataa.
- Tee aina uusi tiedosto datan eri vaiheille.
- Pidä kirjaa eri datan muokkausvaiheista.
    - Minkä vaiheiden kautta päädyit raakadatasta analysoitavaan dataan. Kirjaa eri vaiheet ylös esimerkiksi erilliseen readme (txt) -tiedostoon ja tallenna se samaan kansioon data tiedostojen kanssa.
- Jos organisoit dataasi käyttäen esimerkiksi R tai Python koodia. Kirjaa koodin ylle selkeästi mitä eri vaiheet tekevät. # tämä tekee sitä ja tätä

Esimerkiksi näin:

Teksti mukaillen: Hoyt, P., et al.(2019): Data Carpentry: Data Organization in Spreadsheets for Ecologists, June 2019 (Version v2019.06.2). Zenodo. http://doi.org/10.5281/zenodo.3269869

Pidä kirjaa siitä mitä teet



ResearchPlot

	A	B	C	D	E	F
	ResearchPlot	X	Y	DateCollected	AmountSeeds	
	II A			31/10/1988	18	
	II B			31/10/1988	13	
	II C			31/10/1988	11	
	II D			31/10/1988	11	
	II E			31/10/1988	15	
	II F			31/10/1988	13	
	II H			31/10/1988	11	
	II J			31/10/1988	12	
	II K			31/10/1988	26	
	II L			31/10/1988	10	
	M			31/10/1988	11	
	N			31/10/1988	9	
	O			31/10/1988	16	
	P			31/10/1988	15	

20221115BetulaPubescens_cleaned

README_catacleaningsteps_2022 - Notepad

File Edit Format View Help

Processing notes

From sarvaksen koeala koivunsiemenet.xlxs to 20221115BetulaPubescens_cleaned_V01.csv

2022-11-15 work done ----------

1. Removed header "SARVAKSEN KOEALA KOIVUN SIEMENET"

2. Made two new columns for coordinates, "X" and "Y"

3. Made new column for dates "DateCollected"

4. Named colums that did not have a name "ResearchPlot" and "Amount Seeds"

5. Brought together corresponding date with the amount of seeds cut + paste

6. Removed notes that stated that some years or part of that years data are still in physical notebooks.
   Years mentioned are 2000, 2003, 2005, 2007, 2008, 2009, 2010
   (Or this refers to from what years notebook the data is received)

7. Removed empty cells between rows

8. Saved the file as csv 20221115BetulaPubescens_cleaned_V01.csv

Tehtävä 1

Kouluttaja voi lisätä tähän tarvittavat ohjeet & linkit tehtävän tekoon.

Miten pitää taulukkomuotoinen tiedosto siistinä?

Toimi näin:

1. Laita kaikki muuttujat omiin sarakkeisiin - kuten “lämpötila” tai “paino”
2. Jos haluat laittaa yksikkötiedon, kerro se joko readme-tiedostossa tai laita tämä tieto erilliseen soluun tai muuttujan nimeen jos se on mahdollista
3. Laita jokainen arvo omalle rivilleen.
4. Älä yhdistä usean muuttujan tietoja yhteen soluun. Tämä lisää datan käytettävyyttä. Dataa on esimerkiksi helpompi järjestää kun kaikki muuttujat ovat omissa soluissa.
5. Pidä raakadata raakana - älä koskaan muuta sitä!
6. Tallenna siivottu data tekstimuotoisena tiedostona, kuten CSV (comma-separated values) muodossa. Näin varmistat, että kuka tahansa voi käyttää dataasi. Myös useat repositoriot vaativat avaamaan datasi CSV-muodossa.

Muista! Sarake = Muuttuja, Rivi = Havainto, Solu = Data (arvo).

Teksti mukaillen: Hoyt, P., et al.(2019): Data Carpentry: Data Organization in Spreadsheets for Ecologists, June 2019 (Version v2019.06.2). Zenodo. http://doi.org/10.5281/zenodo.3269869

Sarake = Muuttuja, Rivi = Havainto, Solu = Data (arvo)

	A	B	C	D	E	F	G	H
1	NestID	Species	BZ	Day	Month	Year	Dayofyear	
2	A_001	ACCGEN	HB	19	6	1979	170	
3	A_002	ACCGEN	HB	18	6	2001	169	
4	A_003	ACCGEN	HB	6	7	1984	188	
5	A_004	ACCGEN	HB	26	6	1987	177	
6	A_005	ACCGEN	HB	25	6	1995	176	
7	A_006	ACCGEN	HB	24	6	2010	175	
8	A_007	ACCGEN	HB	21	6	1981	172	
9	A_008	ACCGEN	HB	27	6	1983	178	
10	A_009	ACCGEN	HB	27	6	1986	178	
11	A_010	ACCGEN	HB	26	6	1985	177	
12	A_011	ACCGEN	HB	29	6	2003	180	
13	A_012	ACCGEN	HB	6	7	1992	188	
14	A_013	ACCGEN	HB	17	6	1993	168	
15	A_014	ACCGEN	HB	15	6	1979	166	
16	A_015	ACCGEN	HB	2	7	1980	184	
17	A_016	ACCGEN	HB	18	6	1992	170	

Data Example from: Hällfors, Maria et al. (2020), Data from: Shifts in timing and duration of breeding for 73 boreal bird species over four decades, Dryad, Dataset, https://doi.org/10.5061/dryad.wstqjq2ht

Yksi Välilehti - Yksi Taulukko, Yksi Taulukko - Yksi Välilehti

Useat taulukot yhdellä välilehdellä

- Älä tee useita erillisiä taulukoita yhdelle välilehdelle. Tämä saattaa lukijasta näyttää mukavalle ja käytännölliselle, mutta kone ei tunnista useita erillisiä taulukoita.
- Luomalla useita taulukoita yhdelle välilehdelle luot koneelle vääriä yhteyksiä asioiden välille, sillä kone näkee yhden rivin yhtenä havaintona. Samalla käytät samaa nimeä useissa paikoissa, joka edelleen vaikeuttaa datan siivoamista.

Useat välilehdet taulukossa

- Usein tuntuu myös järkevältä käyttää useita välilehtiä datan organisointiin tai metadatan luontiin. Tämä ei ole täysin kiellettyä tai väärin, mutta se vaikuttaa datan luettavuuteen.
- Jos esimerkiksi teet eri paikoille oman välilehden saattaa se aiheuttaa seuraavanlaisia ongelmia:
- Useiden välilehtien käyttö on haavoittuvaisempi inhimillisille virheille. On liiankin helppoa syöttää dataa väärälle välilehdelle.
- Lisäät yhden ylimääräisen vaiheen ennen kuin pääset analysoimaan dataasi, koska data on joka tapauksessa yhdistettävä yhdeksi taulukoksi. Jos lisäksi välilehdillä on eroavaisuuksia, vaikeuttaa tämä edelleen prosessia ja joudut ehkä manuaalisesti yhdistämään taulukoita, joka lisää mahdollisten virheiden määrää.

Mieti aina onko tarpeellista lisätä uusi välilehti vai voisitko tehdä asian toisella tavalla. Vaikka dataa olisikin tuhansia rivejä, voit aina jäädyttää (freeze) ensimmäisen rivin lukemisen helpottamiseksi.

Teksti mukaillen: Hoyt, P., et al.(2019): Data Carpentry: Data Organization in Spreadsheets for Ecologists, June 2019 (Version v2019.06.2). Zenodo. http://doi.org/10.5281/zenodo.3269869

Muotoilun, kommenttien ja yksiköiden käyttäminen

1. Älä käytä mitään muotoiluja, kuten solujen värittämistä, reunoja, fontteja tiedon ilmaisuun
2. Edellä mainitut eivät ole pysyvää tietoa, eikä kone niitä lue!
3. Jos haluat ilmaista jotain tietoa, luo sille oma sarake
4. Jos haluat lisätä kommentin kertomaan jostain asiasta, älä laita sitä datan joukkoon, vaan luo sille oma sarake.
5. Älä käytä yksiköitä numeerisen tiedon perässä. Kaikkien muuttujien tulisi lähtökohtaisesti olla samoja yksiköitä. Jos ne eivät ole, luo uusi muuttuja ilmaisemaan yksikköä.
6. Älä myöskään käytä erikoismerkkejä jos lisäät tekstiä kommenttikenttään!

Teksti mukaillen: Hoyt, P., et al.(2019): Data Carpentry: Data Organization in Spreadsheets for Ecologists, June 2019 (Version v2019.06.2). Zenodo. http://doi.org/10.5281/zenodo.3269869

Muuttujien nimeäminen

Anna muuttujille kuvaavat nimet!

Muista:

1) Nimet eivät saa sisältää välilyöntejä, numeroita, eikä erikoismerkkejä, koska ne sekoittavat koneen datan lukua.
2) Käytä alaviivaa (_) välilyöntien sijaan
3) Voit käyttää suuria kirjaimia erottamaan sanoja esim. ExampleFileName
4) Älä käytä liian pitkiä nimiä!
5) Kirjaa ylös nimeämiskäytäntö erilliseen metatiedostoon, etenkin jos käytät lyhenteitä, jotta muut käyttäjät ja sinä ymmärtävät mitä muuttujat tarkoittavat

Teksti mukaillen: Hoyt, P., et al.(2019): Data Carpentry: Data Organization in Spreadsheets for Ecologists, June 2019 (Version v2019.06.2). Zenodo. http://doi.org/10.5281/zenodo.3269869

Muuttujien nimeäminen - esimerkki

	A	B	C	D	E	F	G	H
1	NestID	Species	BZ	Day	Month	Year	Dayofyear	
2	A_001	ACCGEN	HB	19	6	1979	170	
3	A_002	ACCGEN	HB	18	6	2001	169	
4	A_003	ACCGEN	HB	6	7	1984	188	
5	A_004	ACCGEN	HB	26	6	1987	177	
6	A_005	ACCGEN	HB	25	6	1995	176	
7	A_006	ACCGEN	HB	24	6	2010	175	
8	A_007	ACCGEN	HB	21	6	1981	172	
9	A_008	ACCGEN	HB	27	6	1983	178	
10	A_009	ACCGEN	HB	27	6	1986	178	
11	A_010	ACCGEN	HB	26	6	1985	177	
12	A_011	ACCGEN	HB	29	6	2003	180	
13	A_012	ACCGEN	HB	6	7	1992	188	
14	A_013	ACCGEN	HB	17	6	1993	168	
15	A_014	ACCGEN	HB	15	6	1979	166	
16	A_015	ACCGEN	HB	2	7	1980	184	
17	A_016	ACCGEN	HB	18	6	1992	170	

Data Example from: Hällfors, Maria et al. (2020), Data from: Shifts in timing and duration of breeding for 73 boreal bird species over four decades, Dryad, Dataset, https://doi.org/10.5061/dryad.wstqjq2ht

Nolla arvojen kirjaamatta jättäminen

Miksi vaivautua kirjaamaan nolla-arvot, kun voit vain jättää solun tyhjäksi?

1) Tyhjä solu ja kirjattu 0 eivät tarkoita samaa asiaa. Koneelle 0 on oikea arvo ja näin ollen dataa, sen sijaan kone tulkitsee tyhjän solun tuntemattomaksi arvoksi (null value)
2) Tilastolliset ohjelmat näin ollen tulkitsevat tyhjäksi jättämäsi solut väärin, eivätkä huomioi näitä analyyseissä. Tämä luonnollisesti aiheuttaa vääristymiä analyyseihin

Teksti mukaillen: Hoyt, P., et al.(2019): Data Carpentry: Data Organization in Spreadsheets for Ecologists, June 2019 (Version v2019.06.2). Zenodo. http://doi.org/10.5281/zenodo.3269869

Puuttuvien arvojen kirjaaminen

Älä käytä 0-arvoja kertomaan puuttuvasta tiedosta!
Nolla kertoo nollahavainnosta tai määreestä.

Mitä tahansa tapaa käytät, kirjaa se ylös metatietoihin!

Table 1. Commonly used null values, limitations, compatibility with common software and a recommendation regarding whether or not it is a good option. Null values are indicated as compatible with specific software if they work consistently and correctly with that software. For example, the null value "NULL" works correctly for certain applications in R, but does not work in others, so it is not presented in the table as R compatible.

Null values	Problems	Compatibility	Recommendation
0	Indistinguishable from a true zero		Never use
Blank	Hard to distinguish values that are missing from those overlooked on entry. Hard to distinguish blanks from spaces, which behave differently.	R, Python, SQL	Best option
-999, 999	Not recognized as null by many programs without user input. Can be inadvertently entered into calculations.		Avoid
NA, na	Can also be an abbreviation (e.g., North America), can cause problems with data type (turn a numerical column into a text column). NA is more commonly recognized than na.	R	Good option
N/A	An alternate form of NA, but often not compatible with software		Avoid
NULL	Can cause problems with data type	SQL	Good option
None	Uncommon. Can cause problems with data type	Python	Avoid
No data	Uncommon. Can cause problems with data type, contains a space		Avoid
Missing	Uncommon. Can cause problems with data type		Avoid
-,+,.	Uncommon. Can cause problems with data type		Avoid

Ethan P White, Elita Baldridge, Zachary T. Brym, Kenneth J. Locey, Daniel J. McGlinn, Sarah R. Supp, Nine simple ways to easier to (re)use your data Vol. 6 No. 2 (2013): Special Issue - Data Sharing in Ecology and Evolution

https://datacarpentry.org/spreadsheet-ecology-lesson/02-common-mistakes/index.html

Teksti mukaillen: Hoyt, P., et al.(2019): Data Carpentry: Data Organization in Spreadsheets for Ecologists, June 2019 (Version v2019.06.2). Zenodo. http://doi.org/10.5281/zenodo.3269869

Metatiedon lisääminen taulukkoon

- Metadatan tuottaminen on elintärkeää aineiston luettavuuden kannalta.
- Metadatan olisi hyvä olla eri tiedostossa, mutta kuitenkin samassa kansiossa datan kanssa
- Metatiedon tulisi sisältää mm. muuttujat, käytetyt yksiköt, tiedot miten tyhjät arvot on koodattu, lyhenteet jne. Toisin sanoen kaikki tarvittava tieto datan lukuun liittyen

Päivämäärät datassa

Käytä päivämääräformaattina (ISO8601): YYYYMMDD (vuosi, kk, päivä), ja aika, jos tarpeen HHMMSS (tunnit, minuutit, sekunnit).

Etenkin ekologiassa käytä YYYY kuvaamaan vuotta.

Merkintä 17 tai 99 voi tarkoittaa:

1917, 2017 (tai jopa vielä vanhempaa)

1899, 1999

	A	B	C	E	F	G	H	I
1	ResearchPlot	X	Y	YEAR	MONTH	DAY	AmountSe	Comment
2	II A			1988	31	10	18	
3	II B			1988	31	10	13	
4	II C			1988	31	10	11	
5	II D			1988	31	10	11	
6	II E			1988	31	10	15	
7	II F			1988	31	10	13	
8	II H			1988	31	10	11	
9	II J			1988	31	10	12	
10	II K			1988	31	10	26	
11	II L			1988	31	10	10	
12	M			1988	31	10	11	
13	N			1988	31	10	9	
14	O			1988	31	10	16	
15	P			1988	31	10	15	
16	R			1988	31	10	15	
17	II A			1991	31	5	24	
18	II B			1991	31	5	0	
19	II C			1991	31	5	1	
20	II D			1991	31	5	3	
21	II E			1991	31	5	80	
22	II F			1991	31	5	9	
23	II H			1991	31	5	99	

Teksti mukaillen: Hoyt, P., et al.(2019): Data Carpentry: Data Organization in Spreadsheets for Ecologists, June 2019 (Version v2019.06.2). Zenodo. http://doi.org/10.5281/zenodo.3269869

Tehtävä 2

Tarkastele annettua excel-taulukkoa

Viitteet

- Slides 6-9
  - MIT School of Engineering Communication Lab;
    - https://mitcommlab.mit.edu/be/commkit/file-structure/#ChooseScaleAimshttps://mitcommlab.mit.edu/be/commkit/file-structure/#ChooseScaleAim
    - https://github.com/mitcommlab/Coding-Documentation/blob/master/File-Structure-Case-Studies.md#case-study-2-a-simple-hierarchy
    - https://github.com/mitcommlab/Coding-Documentation/blob/master/File-Structure-Case-Studies.md#three-examples-of-file-structures-for-different-project-types
  - RDMkit Elixir: https://rdmkit.elixir-europe.org/data_organisation#how-do-you-organise-files-in-a-folder-structure
- Slides 10-13
  - CSC: https://docs.csc.fi/data/datasets/metadata-and-documentation/#data-organization
  - Elixir: https://rdmkit.elixir-europe.org/data_organisation#what-is-the-best-way-to-name-a-file
  - Siiri Fuchs, & Mari Elisa Kuusniemi. (2018). Making a research project understandable - Guide for data documentation (1.2). Zenodo. https://doi.org/10.5281/zenodo.1914401
- Slides 20, 24
  - Data Example from: Hällfors, Maria et al. (2020), Data from: Shifts in timing and duration of breeding for 73 boreal bird species over four decades, Dryad, Dataset, https://doi.org/10.5061/dryad.wstqjq2ht
- Slides 14, 16, 19, 21-23, 25-28
  - **The Carpentries.** Text mainly adapted from online course Data Organization in Spreadsheets for Ecologists licensed under CC-BY 4.0 2018–2023 by The Carpentries
    - Peter R. Hoyt, Christie Bahlai, Tracy K. Teal (Eds.), Erin Alison Becker, Aleksandra Pawlik, Peter Hoyt, Francois Michonneau, Christie Bahlai, Toby Reiter, et al. (2019, July 5). datacarpentry/spreadsheet-ecology-lesson: Data Carpentry: Data Organization in Spreadsheets for Ecologists, June 2019 (Version v2019.06.2). Zenodo. http://doi.org/10.5281/zenodo.3269869
    - https://datacarpentry.org/spreadsheet-ecology-lesson/
    - https://datacarpentry.org/spreadsheet-ecology-lesson/02-common-mistakes/index.html
  - EDINA and Data Library, University of Edinburgh: https://rdmkit.elixir-europe.org/data_organisation#what-is-the-best-way-to-name-a-file
  - Tidy data: https://www.jstatsoft.org/article/view/v059i10
  - Data organization in spreadsheets: https://peerj.com/preprints/3183/
  - Ethan P White, Elita Baldridge, Zachary T. Brym, Kenneth J. Locey, Daniel J. McGlinn, Sarah R. Supp, Nine simple ways to make it easier to (re)use your data Vol. 6 No. 2 (2013): Special Issue - Data Sharing in Ecology and Evolution