pyLabbook Manager Interface

Quick Start

You need: python3.6+, numpy, pandas, sqlite3, xlrd, openpyxl, tkinter

Ideally: pandas 0.24.0+, but you can manage with lower version numbers.

1. Obtain a full pyLabbook distribution by cloning from https://github.com/Nhydrazine/pyLabbook
2. Open a terminal window
3. Go to the pyLabbook root folder using `cd`
4. Run the testme.py script with `python testme.py`

```
sys .................................... OK
python ................................. 3.6.5

Checking for Required Modules
os ..................................... OK
re ..................................... OK
shutil ................................. OK
numpy .................................. OK
pandas ................................. OK
sqlite3 ................................ OK
xlrd ................................... OK
openpyxl ............................... OK

Checking for Manager Modules
tkinter ................................ OK
tkinter.messagebox ..................... OK
tkinter.scrolledtext ................... OK
tkinter.filedialog ..................... OK
tkinter.ttk ............................ OK

Checking for Manager Widgets
guiWidgets ............................. OK
ttk.Frame .............................. OK
ttk.Label .............................. OK
ttk.Entry .............................. OK
ttk.Button ............................. OK
ttk.Checkbutton ........................ OK
ttk.Combobox ........................... OK
ttk.Treeview ........................... OK

Checking pyLabbook distribution at /Users/nic/github/pyLabbook
/Users/nic/github/pyLabbook/python/pyLabbook/labbooks  OK
/Users/nic/github/pyLabbook/python/PyLabbook/protocols  OK
/Users/nic/github/pyLabbook/databases  OK
/Users/nic/github/pyLabbook/repositories  OK
/Users/nic/github/pyLabbook/exports  OK
/Users/nic/github/pyLabbook/imports  OK
/Users/nic/github/pyLabbook/python/pyLabbook/pyLabbook.py  OK
/Users/nic/github/pyLabbook/python/pyLabbook/pyLabbook.py  OK
/Users/nic/github/pyLabbook/python/pyLabbook/core.py  OK
/Users/nic/github/pyLabbook/python/pyLabbook/core.py  OK
/Users/nic/github/pyLabbook/python/operations.py  OK

Checking pyLabbook modules and classes
pyLabbook .............................. 0.b.1
pyLabbook.pyLabbook .................... OK
pyLabbook.pyProtocol ................... OK
pyLabbook.core ......................... OK
pyLabbook.SQLEngines.engine ............ OK
pyLabbook.SQLEngines.manager ........... OK
SQLITE3 engine ......................... OK
Warning: you may experience some odd behavior realted to:
--> update pandas to version 0.24.0 or higher
Done
```

These are required for the manager to work properly.

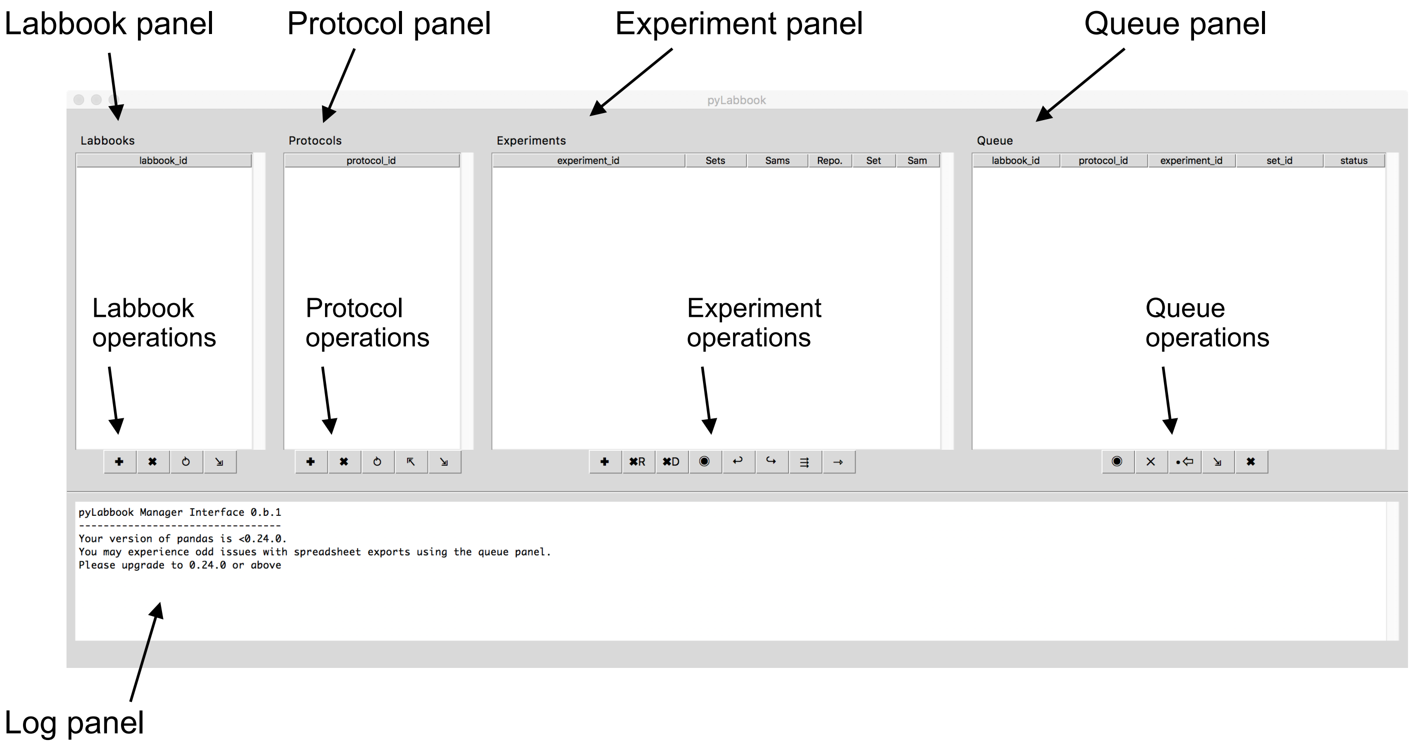
If missing, you can either:
(a) install a newer version of python
(b) install yourself: https://tkdocs.com/tutorial/install.html

5. If all good, run the manager script with `python manager.py`

Missing modules can be installed with `pip`, `conda`, or whatever you use to install python modules.

pyLabbook Manager Interface

Main Window



```
pyLabbook Manager Interface 0.b.1
---------------------------------
Your version of pandas is <0.24.0.
You may experience odd issues with spreadsheet exports using the queue panel.
Please upgrade to 0.24.0 or above
```

Labbook panel

Displays the labbooks that you have created or installed.

Protocol panel

Displays the protocols that you have created or installed.

Experiment panel

Displays the experiments that belong to a targeted labbook and a targeted protocol.

Queue panel

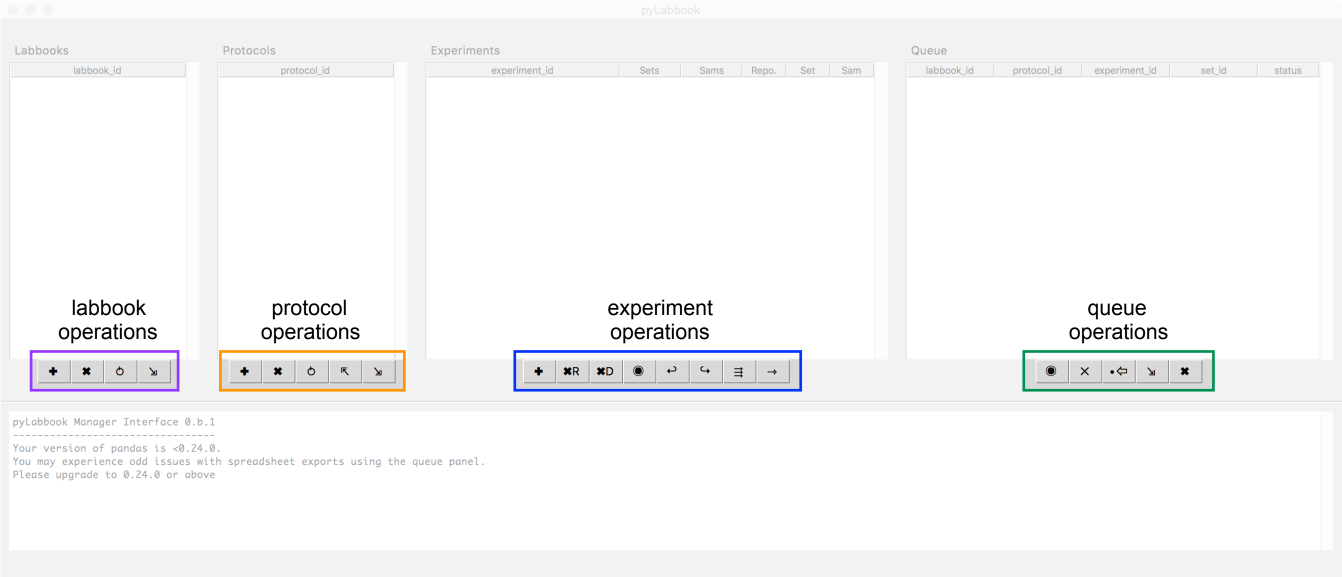
Displays experiment/set identities for data to be transferred or exported.

Log panel

Report of actions and results/status.

pyLabbook Manager Interface

Main Window



```
pyLabbook Manager Interface 0.b.1
--------------------------------
Your version of pandas is <0.24.0.
You may experience odd issues with spreadsheet exports using the queue panel.
Please upgrade to 0.24.0 or above
```

✚ **Create Labbook** with or without importing an existing database.

! ✖ **Delete Labbook** that is highlighted.

⥁ **Refresh** list of labbooks.

⇲ **Export** the database of the highlighted labbook.

✚ **Create Protocol**.

! ✖ **Delete Protocol** that is highlighted.

⥁ **Refresh** protocol list.

⇱ **Import** protocol.

⇲ **Export** protocol.

✚ **Initialize** new experiment.

! ✖R **Delete** experiment repository.

! ✖D **Delete** experiment database records.

◉ **Target** the highlighted labbook and protocol.

↩ **Store** repository data into database.

↪ **ReStore** repository data from database.

⇶ **Queue all** displayed experiments.

→ **Queue highlighted** experiments.

◉ **Target** highlighted labbook.

✕ **Clear** queue list.

•⇦ **Transfer** queued data to targeted labbook database.

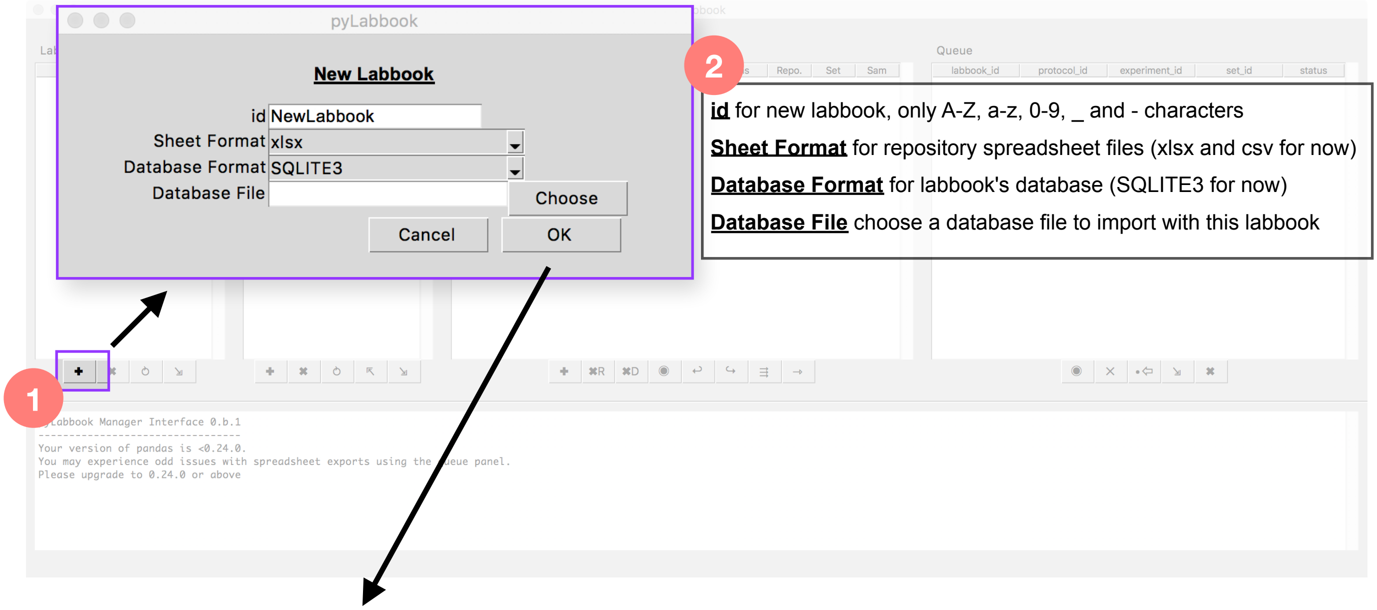
⇲ **Export** queued data to spreadsheet files.

✖ **Remove** highlighted item from queue.

! These operations result in the loss of data and cannot be undone.

pyLabbook Manager Interface

Creating a Labbook



New Labbook

id NewLabbook
Sheet Format xlsx
Database Format SQLITE3
Database File

```
pyLabbook Manager Interface 0.b.1
----------------------------------
Your version of pandas is <0.24.0.
You may experience odd issues with spreadsheet exports using the queue panel.
Please upgrade to 0.24.0 or above
```

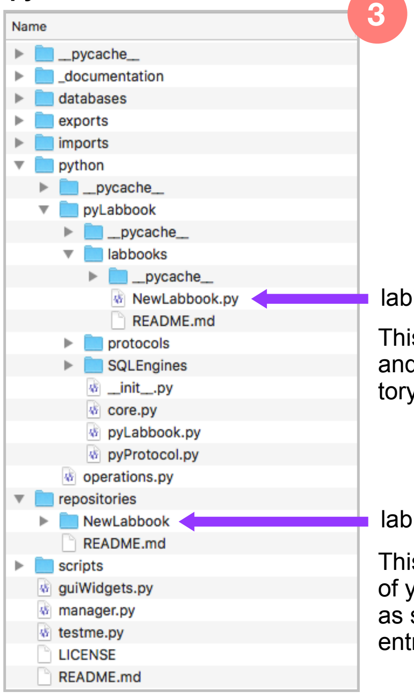
id for new labbook, only A-Z, a-z, 0-9, _ and - characters

Sheet Format for repository spreadsheet files (xlsx and csv for now)

Database Format for labbook's database (SQLITE3 for now)

Database File choose a database file to import with this labbook

pyLabbook distribution folder



```
Name
▶ __pycache__
▶ _documentation
▶ databases
▶ exports
▶ imports
▼ python
    ▶ __pycache__
    ▼ pyLabbook
        ▶ __pycache__
        ▼ labbooks
            ▶ __pycache__
              NewLabbook.py
              README.md
        ▶ protocols
        ▶ SQLEngines
          __init__.py
          core.py
          pyLabbook.py
          pyProtocol.py
      operations.py
▼ repositories
    ▶ NewLabbook
      README.md
▶ scripts
  guiWidgets.py
  manager.py
  testme.py
  LICENSE
  README.md
```

labbook python file in python/pyLabbook/labbooks/

This python file extends the pyLabbook.pyLabbook class and describes the labbook and the locations of its repository and database files.

labbook repository folder in repositories/

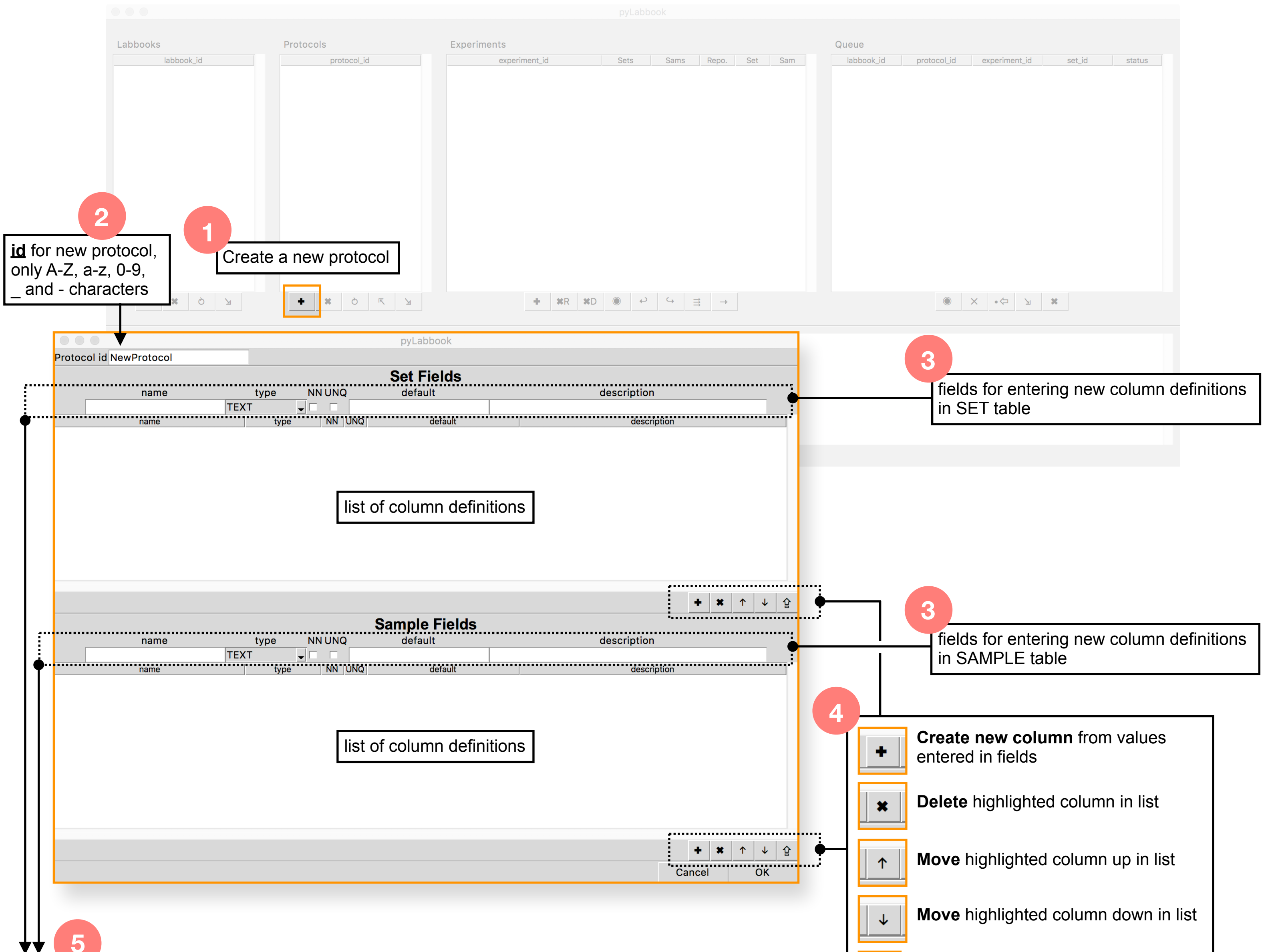
This folder contains a substructure that organizes all of your experimental data by protocol and experiment as spreadsheet files. These files are intended for data entry purposes and for temporary access to your data.

NOTE: Database file and additional repository structure will appear when data is added to the labbook.

NOTE: You will only have access to data from an imported database when you obtain the corresponding protocol file.

pyLabbook Manager Interface

Creating a Protocol



Protocol id NewProtocol

1 Create a new protocol

2 **id** for new protocol, only A-Z, a-z, 0-9, _ and - characters

3 fields for entering new column definitions in SET table

list of column definitions

3 fields for entering new column definitions in SAMPLE table

list of column definitions

4

✚ **Create new column** from values entered in fields

✖ **Delete** highlighted column in list

↑ **Move** highlighted column up in list

↓ **Move** highlighted column down in list

⇧ **Pop/edit** highlighted column from list, back into entry fields for editing

5

Field Properties (applies to both SET and SAMPLE tables)

name the name of the field, A-Z, a-z, 0-9, _ and - characters only.

type type of data allowed in field:

- TEXT includes all characters and can be any length.
- DATE is treated as TEXT.
- INTEGER is any whole integer number.
- REAL is any floating point number.
- NUMERIC is treated as a REAL.

NN stands for **not null**. If checked, every record must supply a value for this field.

UNQ stands for **unique**. If checked, every record must supply a unique value for this field.

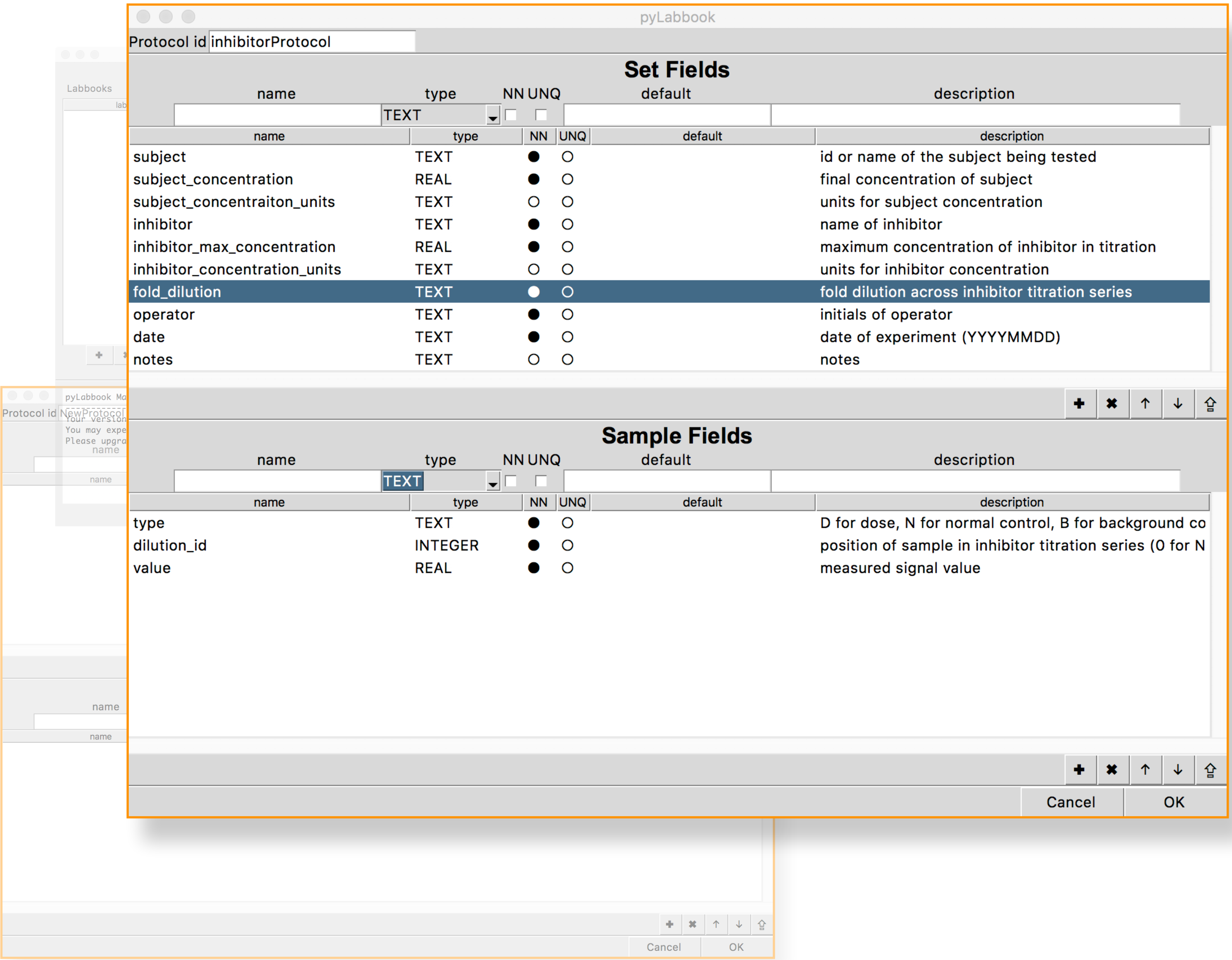
Default is a default value to use when no value is provided. This must be compatible with the field's **type**.

Description is used to describe the field and/or its values.

pyLabbook Manager Interface

Creating a Protocol

Specify the SET and SAMPLE table fields



Protocol id inhibitorProtocol

Set Fields

name | type TEXT | NN □ | UNQ □ | default | description

name	type	NN	UNQ	default	description
subject	TEXT	●	○		id or name of the subject being tested
subject_concentration	REAL	●	○		final concentration of subject
subject_concentraiton_units	TEXT	○	○		units for subject concentration
inhibitor	TEXT	●	○		name of inhibitor
inhibitor_max_concentration	REAL	●	○		maximum concentration of inhibitor in titration
inhibitor_concentration_units	TEXT	○	○		units for inhibitor concentration
fold_dilution	TEXT	●	○		fold dilution across inhibitor titration series
operator	TEXT	●	○		initials of operator
date	TEXT	●	○		date of experiment (YYYYMMDD)
notes	TEXT	○	○		notes

✚ ✖ ↑ ↓ ⇪

Sample Fields

name | type TEXT | NN □ | UNQ □ | default | description

name	type	NN	UNQ	default	description
type	TEXT	●	○		D for dose, N for normal control, B for background co
dilution_id	INTEGER	●	○		position of sample in inhibitor titration series (0 for N
value	REAL	●	○		measured signal value

✚ ✖ ↑ ↓ ⇪

Cancel | OK

- **Order of fields** is the order that they will appear in spreadsheets, database displays and python interfaces. The order of the fields **will not effect your data**, but odd orders can make data entry cumbersome. Be sure to take a moment to order your fields in an intuitive way.

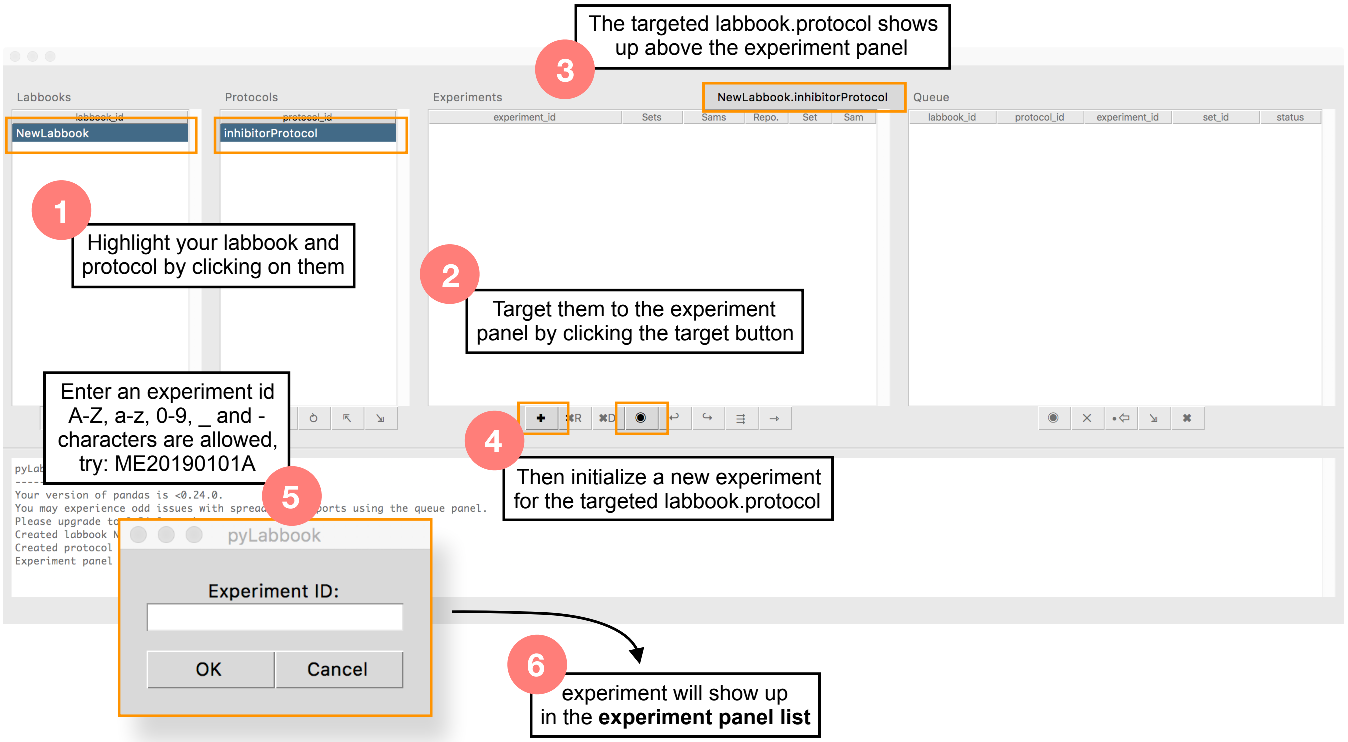
- **Protocols are permanent** and cannot be edited once created. *Take the time to think about your data structure design before committing data to it - **it is worth it**.*

- **You only need to create a protocol once**. If carefully planned you may never have to think about the structure again. So take the time to do it right.

- **Data can be mapped to new protocols**. In the event that you really do need to add or alter the structure of a protocol's data, you can use the python interface to map your old data into a new protocol structure.

pyLabbook Manager Interface

Initialize an Experiment



3 The targeted labbook.protocol shows up above the experiment panel

1 Highlight your labbook and protocol by clicking on them

2 Target them to the experiment panel by clicking the target button

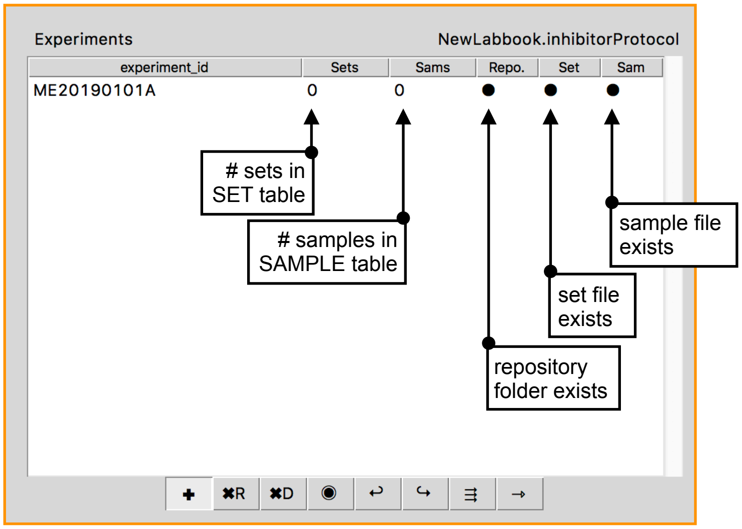
4 Then initialize a new experiment for the targeted labbook.protocol

Enter an experiment id A-Z, a-z, 0-9, _ and - characters are allowed, try: ME20190101A

5

```
pyLab
Your version of pandas is <0.24.0.
You may experience odd issues with spread        ports using the queue panel.
Please upgrade to
Created labbook N
Created protocol
Experiment panel
```

6 experiment will show up in the **experiment panel list**



experiment_id	Sets	Sams	Repo.	Set	Sam
ME20190101A	0	0	●	●	●

# sets in SET table

# samples in SAMPLE table

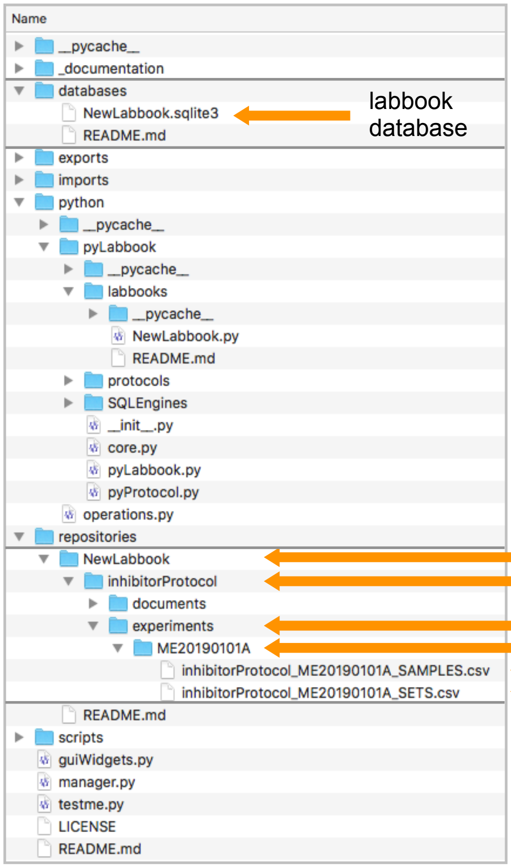
repository folder exists

set file exists

sample file exists

7 experiment repository files have also been created

pyLabbook distribution folder



```
Name
▶ __pycache__
▶ _documentation
▼ databases
    NewLabbook.sqlite3
    README.md
▶ exports
▶ imports
▼ python
  ▶ __pycache__
  ▼ pyLabbook
    ▶ __pycache__
    ▼ labbooks
      ▶ __pycache__
        NewLabbook.py
        README.md
    ▶ protocols
    ▶ SQLEngines
      __init__.py
      core.py
      pyLabbook.py
      pyProtocol.py
    operations.py
▼ repositories
  ▼ NewLabbook
    ▼ inhibitorProtocol
      ▶ documents
      ▼ experiments
        ▼ ME20190101A
            inhibitorProtocol_ME20190101A_SAMPLES.csv
            inhibitorProtocol_ME20190101A_SETS.csv
    README.md
▶ scripts
  guiWidgets.py
  manager.py
  testme.py
  LICENSE
  README.md
```

labbook database

NewLabbook's repository for inhibitorProtocol files

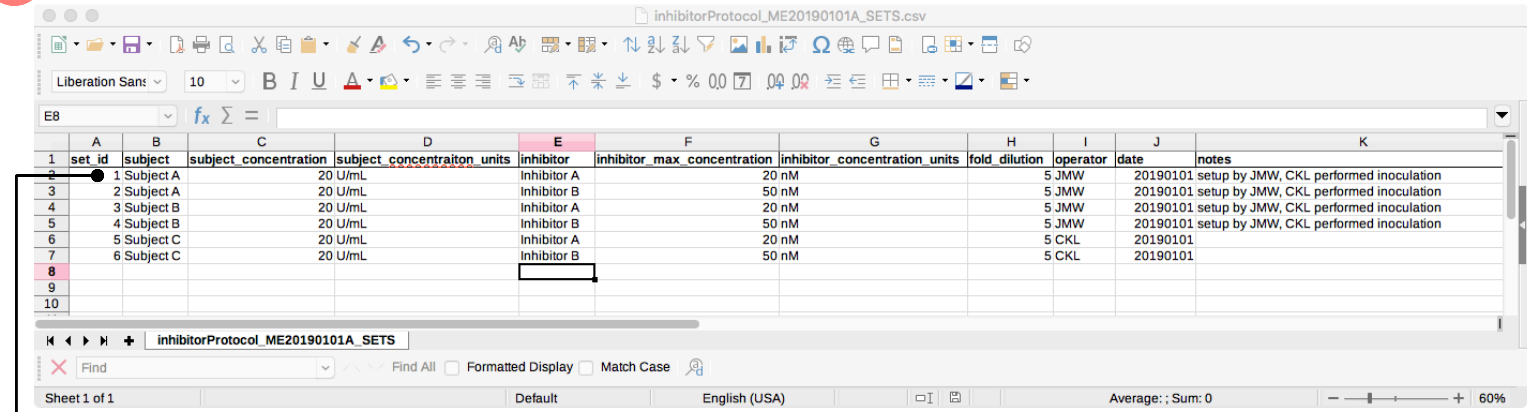
experiment folders for inhibitorProtocol experiments

set and sample files for inhibitorProtocol, experiment ME20190101A

pyLabbook Manager Interface

Data Entry

1 Using your favorite spreadsheet editor, open the SETS file and and enter relevant experimental conditions and information



inhibitorProtocol_ME20190101A_SETS.csv

E8

	A	B	C	D	E	F	G	H	I	J	K
1	set_id	subject	subject_concentration	subject_concentraiton_units	inhibitor	inhibitor_max_concentration	inhibitor_concentration_units	fold_dilution	operator	date	notes
2	1	Subject A	20	U/mL	Inhibitor A	20	nM	5	JMW	20190101	setup by JMW, CKL performed inoculation
3	2	Subject A	20	U/mL	Inhibitor B	50	nM	5	JMW	20190101	setup by JMW, CKL performed inoculation
4	3	Subject B	20	U/mL	Inhibitor A	20	nM	5	JMW	20190101	setup by JMW, CKL performed inoculation
5	4	Subject B	20	U/mL	Inhibitor B	50	nM	5	JMW	20190101	setup by JMW, CKL performed inoculation
6	5	Subject C	20	U/mL	Inhibitor A	20	nM	5	CKL	20190101	
7	6	Subject C	20	U/mL	Inhibitor B	50	nM	5	CKL	20190101	
8											
9											
10											

inhibitorProtocol_ME20190101A_SETS

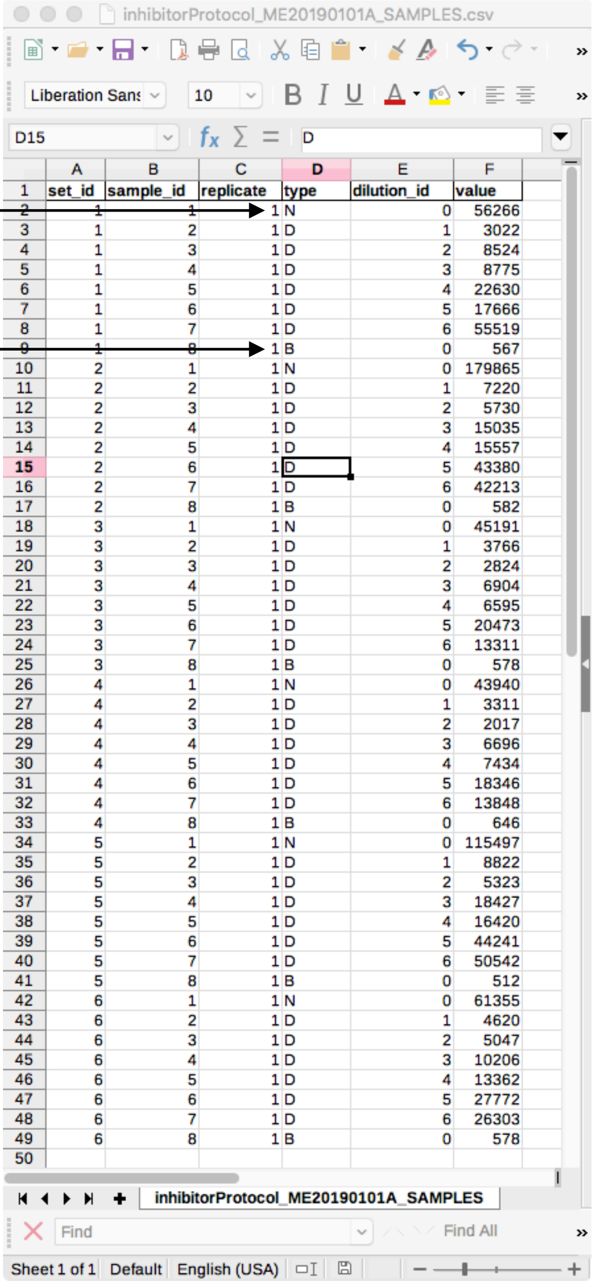
2 Open the SAMPLES file and fill in experimental measurement results. Use cut/paste strategies described later for rapid formatting, or use python script(s) to automatically process instrument result files.

All data for set 1
a complete dose-effect curve for Subject A with Inhibitor A

N Normal control (activity without inhibitor)

D Dosed samples with decreasing inhibitor concentrations

B Background control (signal without activity)



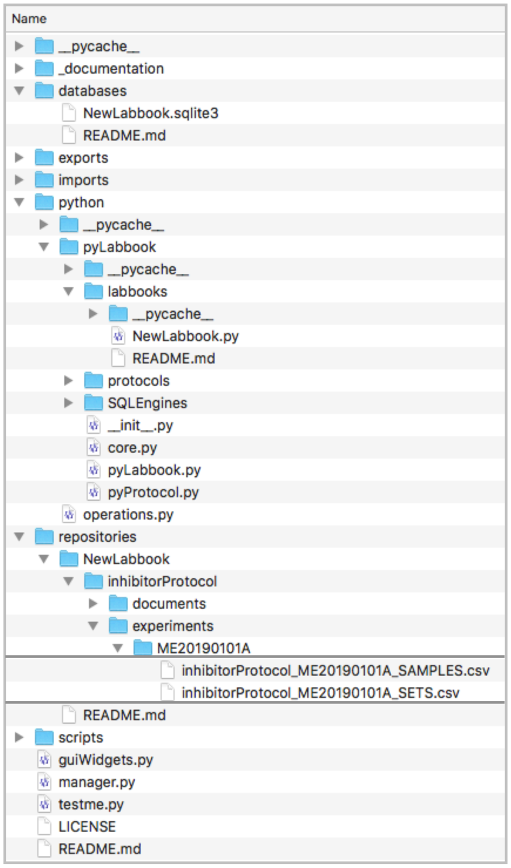
inhibitorProtocol_ME20190101A_SAMPLES.csv

D15 D

	A	B	C	D	E	F
1	set_id	sample_id	replicate	type	dilution_id	value
2	1	1	1	N	0	56266
3	1	2	1	D	1	3022
4	1	3	1	D	2	8524
5	1	4	1	D	3	8775
6	1	5	1	D	4	22630
7	1	6	1	D	5	17666
8	1	7	1	D	6	55519
9	1	8	1	B	0	567
10	2	1	1	N	0	179865
11	2	2	1	D	1	7220
12	2	3	1	D	2	5730
13	2	4	1	D	3	15035
14	2	5	1	D	4	15557
15	2	6	1	D	5	43380
16	2	7	1	D	6	42213
17	2	8	1	B	0	582
18	3	1	1	N	0	45191
19	3	2	1	D	1	3766
20	3	3	1	D	2	2824
21	3	4	1	D	3	6904
22	3	5	1	D	4	6595
23	3	6	1	D	5	20473
24	3	7	1	D	6	13311
25	3	8	1	B	0	578
26	4	1	1	N	0	43940
27	4	2	1	D	1	3311
28	4	3	1	D	2	2017
29	4	4	1	D	3	6696
30	4	5	1	D	4	7434
31	4	6	1	D	5	18346
32	4	7	1	D	6	13848
33	4	8	1	B	0	646
34	5	1	1	N	0	115497
35	5	2	1	D	1	8822
36	5	3	1	D	2	5323
37	5	4	1	D	3	18427
38	5	5	1	D	4	16420
39	5	6	1	D	5	44241
40	5	7	1	D	6	50542
41	5	8	1	B	0	512
42	6	1	1	N	0	61355
43	6	2	1	D	1	4620
44	6	3	1	D	2	5047
45	6	4	1	D	3	10206
46	6	5	1	D	4	13362
47	6	6	1	D	5	27772
48	6	7	1	D	6	26303
49	6	8	1	B	0	578
50						

inhibitorProtocol_ME20190101A_SAMPLES

pyLabbook distribution folder



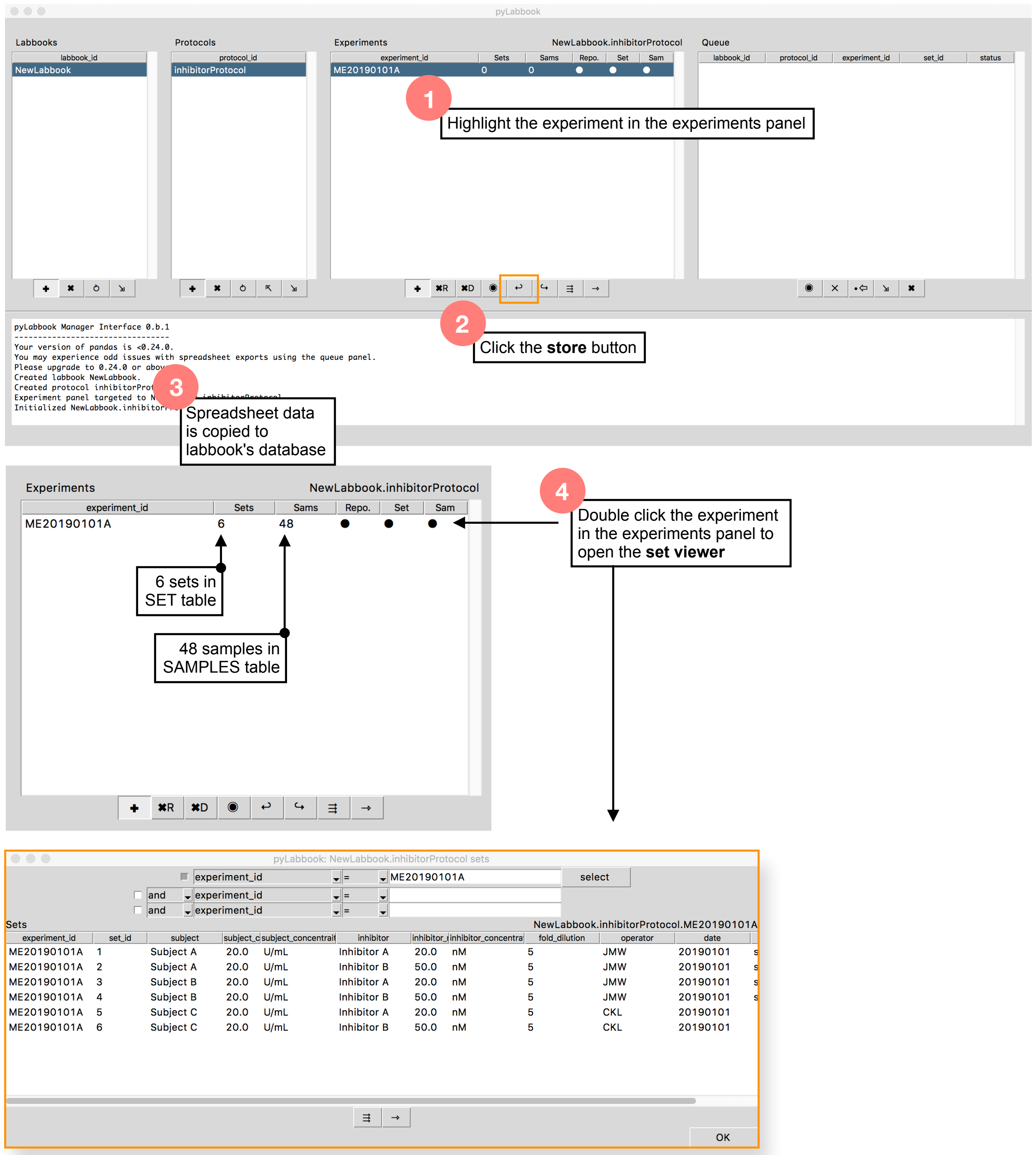
Name

- __pycache__
- _documentation
- databases
  - NewLabbook.sqlite3
  - README.md
- exports
- imports
- python
  - __pycache__
  - pyLabbook
    - __pycache__
    - labbooks
      - __pycache__
      - NewLabbook.py
      - README.md
    - protocols
    - SQLEngines
    - __init__.py
    - core.py
    - pyLabbook.py
    - pyProtocol.py
  - operations.py
- repositories
  - NewLabbook
    - inhibitorProtocol
      - documents
      - experiments
        - ME20190101A
          - inhibitorProtocol_ME20190101A_SAMPLES.csv
          - inhibitorProtocol_ME20190101A_SETS.csv
  - README.md
- scripts
- guiWidgets.py
- manager.py
- testme.py
- LICENSE
- README.md

images show are libre office calc, a free, open source spreadsheet editor.

pyLabbook Manager Interface

Data Storage



1 Highlight the experiment in the experiments panel

2 Click the **store** button

```
pyLabbook Manager Interface 0.b.1
-----------------------------------
Your version of pandas is <0.24.0.
You may experience odd issues with spreadsheet exports using the queue panel.
Please upgrade to 0.24.0 or abov
Created labbook NewLabbook.
Created protocol inhibitorProt
Experiment panel targeted to N                inhibitorProtocol
Initialized NewLabbook.inhibitorP
```

3 Spreadsheet data is copied to labbook's database

Experiments NewLabbook.inhibitorProtocol

experiment_id	Sets	Sams	Repo.	Set	Sam
ME20190101A	6	48	●	●	●

6 sets in SET table

48 samples in SAMPLES table

4 Double click the experiment in the experiments panel to open the **set viewer**

pyLabbook: NewLabbook.inhibitorProtocol sets

■ experiment_id = ME20190101A select

□ and experiment_id =

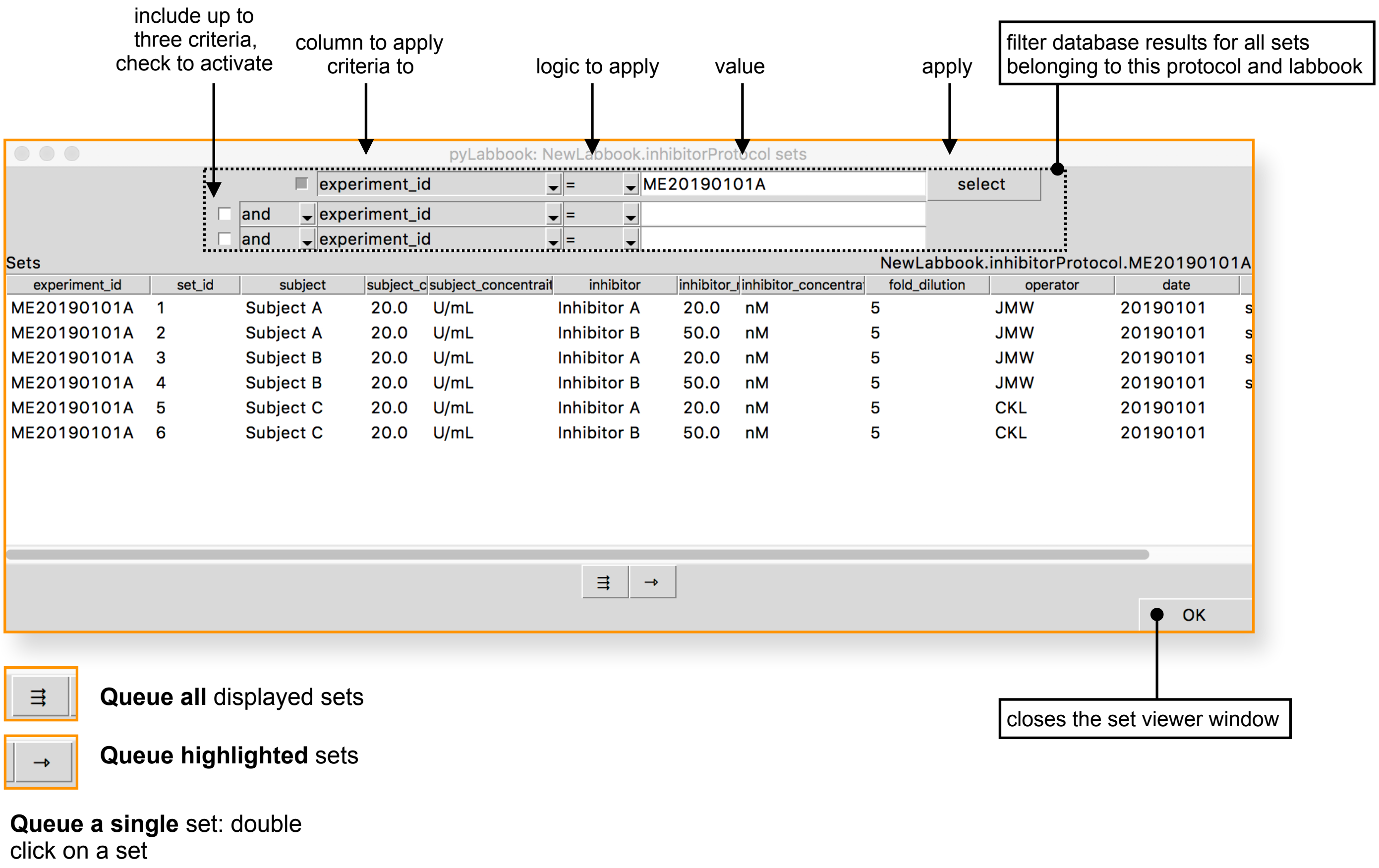
□ and experiment_id =

Sets NewLabbook.inhibitorProtocol.ME20190101A

experiment_id	set_id	subject	subject_c	subject_concentrat	inhibitor	inhibitor_	inhibitor_concentra	fold_dilution	operator	date	
ME20190101A	1	Subject A	20.0	U/mL	Inhibitor A	20.0	nM	5	JMW	20190101	s
ME20190101A	2	Subject A	20.0	U/mL	Inhibitor B	50.0	nM	5	JMW	20190101	s
ME20190101A	3	Subject B	20.0	U/mL	Inhibitor A	20.0	nM	5	JMW	20190101	s
ME20190101A	4	Subject B	20.0	U/mL	Inhibitor B	50.0	nM	5	JMW	20190101	s
ME20190101A	5	Subject C	20.0	U/mL	Inhibitor A	20.0	nM	5	CKL	20190101	
ME20190101A	6	Subject C	20.0	U/mL	Inhibitor B	50.0	nM	5	CKL	20190101	

pyLabbook Manager Interface

Set Viewer



include up to three criteria, check to activate

column to apply criteria to

logic to apply

value

apply

filter database results for all sets belonging to this protocol and labbook

pyLabbook: NewLabbook.inhibitorProtocol sets

☐		experiment_id	=	ME20190101A	select
☐	and	experiment_id	=		
☐	and	experiment_id	=		

Sets

NewLabbook.inhibitorProtocol.ME20190101A

experiment_id	set_id	subject	subject_c	subject_concentrati	inhibitor	inhibitor_i	inhibitor_concentra	fold_dilution	operator	date	
ME20190101A	1	Subject A	20.0	U/mL	Inhibitor A	20.0	nM	5	JMW	20190101	s
ME20190101A	2	Subject A	20.0	U/mL	Inhibitor B	50.0	nM	5	JMW	20190101	s
ME20190101A	3	Subject B	20.0	U/mL	Inhibitor A	20.0	nM	5	JMW	20190101	s
ME20190101A	4	Subject B	20.0	U/mL	Inhibitor B	50.0	nM	5	JMW	20190101	s
ME20190101A	5	Subject C	20.0	U/mL	Inhibitor A	20.0	nM	5	CKL	20190101	
ME20190101A	6	Subject C	20.0	U/mL	Inhibitor B	50.0	nM	5	CKL	20190101	

closes the set viewer window

⇶ **Queue all** displayed sets

→ **Queue highlighted** sets

Queue a single set: double click on a set

- **Current logic is based on sqlite**. This includes the *like* logical operator where you can use the % symbol to specify "any character at this position" and the _ symbol to indicate "anything of any length here". For example, %ello will match "hello", "mello" and "yello", while "hel_" will match any word or phrase that beings with the letters "hel".

- **Logic and values should be consistent with field data type**. Specifying a > or < operator on a TEXT field will result in unexpected results or errors.

- **The set viewer does not edit data**. The set viewer is not intended to edit the values in the database. The database is intended to be a permanent record. To do this, you can edit the repository file(s) and **store** them again to overwrite, or you can use a sqlite database GUI (I recommend SQLiteStudio).

pyLabbook Manager Interface

TO BE COMPLETED