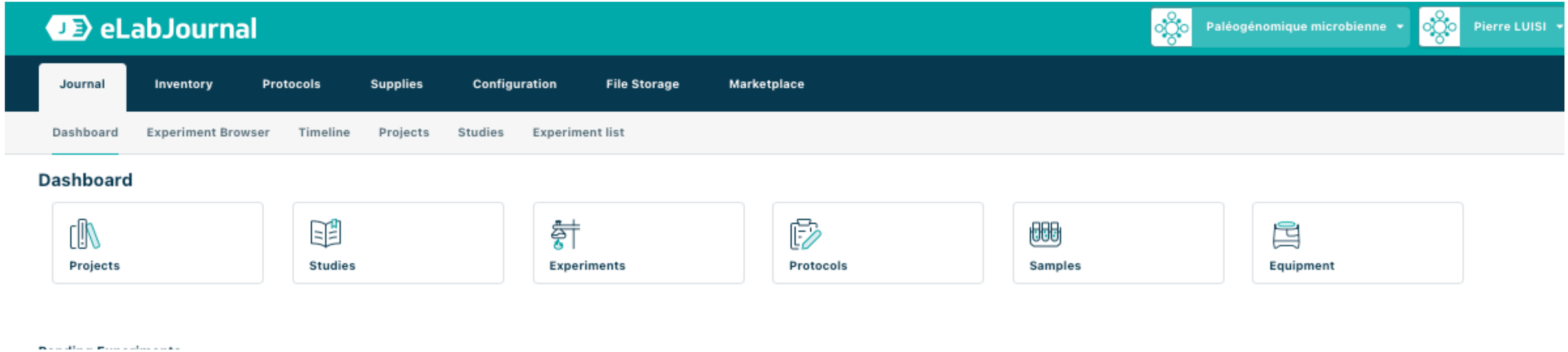


API for downloading data from eLab

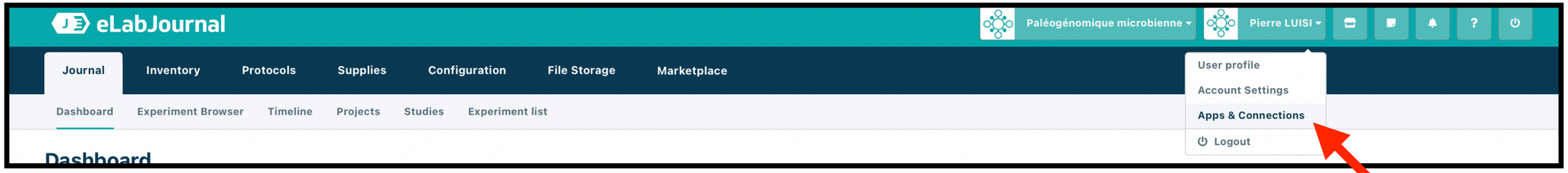
https://github.com/pierrespc/eLab/tree/main/DownloadData/FromFilter

Metapaleogenomics Lab. Institut Pasteur. 18/11/2021

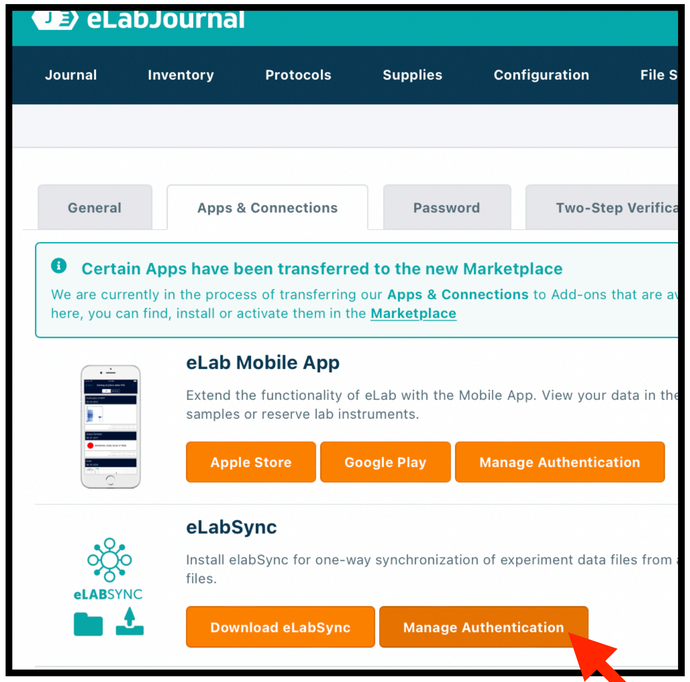
MICROBIAL PALEOGENOMICS

Generate Token

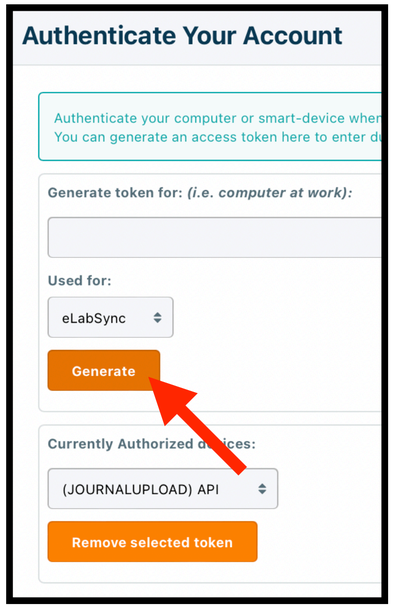
1.



2.



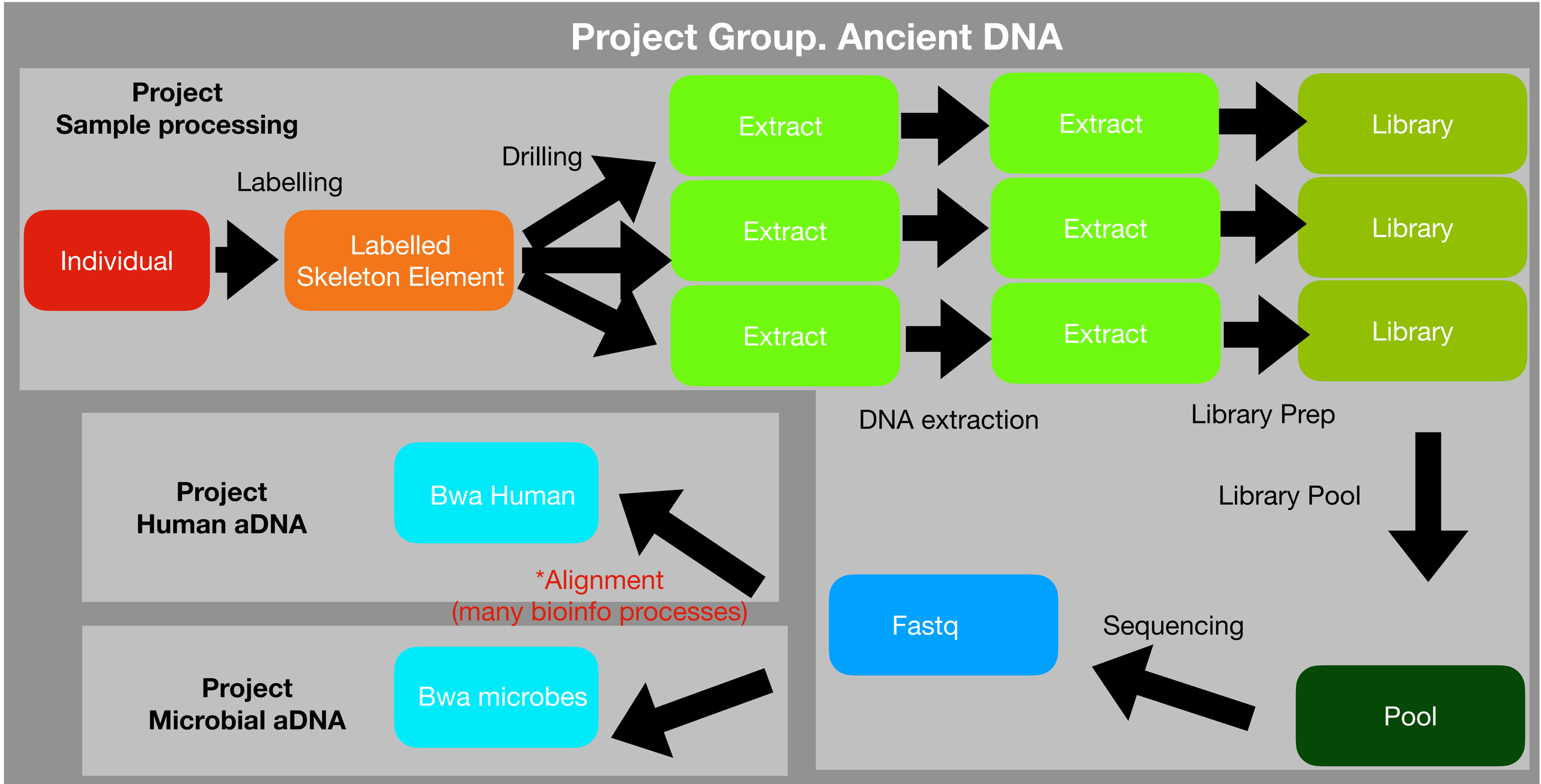
3.



4.

Copy the token (after the semicolon) and save file to a plain file

eLab organization



API for downloading data applying filters

A bottom-up approach

- Start with the lowest level of sample type for which a filter is required
  ==> retrieve IDs and parent IDs

- Go iteratively to the next levels:
  on IDs retrieved at previous step (that is the parent IDs from previous step): apply filter for that level and get the parent IDs

- it is a AND logic function across Sample type levels: we output only entries for which the whole lineage fulfils the conditions at all levels tested

A branched set-up from Archaeological Sites to Genomic Data files

API for downloading data applying filters

A bottom-up approach

- It is a AND logic function across Sample type levels:
  we output only entries for which the whole lineage fulfils the conditions at all levels tested

Extract ID	Quantity	Extract Type	parentID
AR0001.1.01	10mg	Pulp	AR0001.1
AR0001.1.02	10mg	Root apex	AR0001.1
AR2222.1.01	100mg	Root	AR2222.1
AR2222.1.02	4mg	Root Apex	AR2222.1
AR9999.1.01	45mg	Root	AR9999.1
AR9999.2.01	30mg	Root	AR9999.2

API for downloading data applying filters

A bottom-up approach

- It is a AND logic function across Sample type levels:
  we output only entries for which the whole lineage fulfils the conditions at all levels tested

Root or Root Apex
More than 5 mg

Extract ID	Quantity	Extract Type	parentID
~~AR0001.1.01~~	~~10mg~~	~~Pulp~~	~~AR0001.1~~
AR0001.1.02	10mg	Root apex	AR0001.1
AR2222.1.01	100mg	Root	AR2222.1
~~AR2222.1.02~~	~~4mg~~	~~Root Apex~~	~~AR2222.1~~
AR9999.1.01	45mg	Root	AR9999.1
AR9999.2.01	30mg	Root	AR9999.2

API for downloading data applying filters

A bottom-up approach

- It is a AND logic function across Sample type levels:
  we output only entries for which the whole lineage fulfils the conditions at all levels tested

Root or Root Apex
More than 5 mg

Ske Ele ID	Date Labelling	parentID
AR0001.1	2021-04-12	AR0001.1
AR2222.1	2021-05-22	AR0001.1
~~**AR5555.1**~~	~~2021-05-22~~	~~AR5555~~
AR9999.1	2021-11-01	AR9999.1
AR9999.2	2021-04-12	AR9999.2

Extract ID	Quantity	Extract Type	parentID
~~**AR0001.1.01**~~	~~10mg~~	~~Pulp~~	~~AR0001.1~~
AR0001.1.02	10mg	Root apex	AR0001.1
AR2222.1.01	100mg	Root	AR2222.1
~~**AR2222.1.02**~~	~~4mg~~	~~Root Apex~~	~~AR2222.1~~
AR9999.1.01	45mg	Root	AR9999.1
AR9999.2.01	30mg	Root	AR9999.2

API for downloading data applying filters

A bottom-up approach

- It is a AND logic function across Sample type levels:
  we output only entries for which the whole lineage fulfils the conditions at all levels tested

Labelled
before November 2021

Ske Ele ID	Date Labelling	parentID
AR0001.1	2021-04-12	AR0001.1
AR2222.1	2021-05-22	AR0001.1
~~**AR5555.1**~~	~~2021-05-22~~	~~AR5555~~
~~**AR9999.1**~~	~~2021-11-01~~	~~AR9999.1~~
AR9999.2	2021-04-12	AR9999.2

Root or Root Apex
More than 5 mg

Extract ID	Quantity	Extract Type	parentID
~~**AR0001.1.01**~~	~~10mg~~	~~Pulp~~	~~AR0001.1~~
AR0001.1.02	10mg	Root apex	AR0001.1
AR2222.1.01	100mg	Root	AR2222.1
~~**AR2222.1.02**~~	~~4mg~~	~~Root Apex~~	~~AR2222.1~~
AR9999.1.01	45mg	Root	AR9999.1
AR9999.2.01	30mg	Root	AR9999.2

API for downloading data applying filters

A bottom-up approach

- It is a AND logic function across Sample type levels:
  we output only entries for which the whole lineage fulfils the conditions at all levels tested

Ind ID	Archaeologist group	parentID
AR0001	Fulano	Site A
AR2222	Jaimito	Site G
~~**AR5555**~~	~~Jaimito~~	~~Site G~~
AR9999	Jaimito	Site Z

Labelled
before November 2021

Ske Ele ID	Date Labelling	parentID
AR0001.1	2021-04-12	AR0001.1
AR2222.1	2021-05-22	AR0001.1
~~**AR5555.1**~~	~~2021-05-22~~	~~AR5555~~
~~**AR9999.1**~~	~~2021-11-01~~	~~AR9999.1~~
AR9999.2	2021-04-12	AR9999.2

Root or Root Apex
More than 5 mg

Extract ID	Quantity	Extract Type	parentID
~~**AR0001.1.01**~~	~~10mg~~	~~Pulp~~	~~AR0001.1~~
AR0001.1.02	10mg	Root apex	AR0001.1
AR2222.1.01	100mg	Root	AR2222.1
~~**AR2222.1.02**~~	~~4mg~~	~~Root Apex~~	~~AR2222.1~~
AR9999.1.01	45mg	Root	AR9999.1
AR9999.2.01	30mg	Root	AR9999.2

API for downloading data applying filters

A bottom-up approach

- It is a AND logic function across Sample type levels:
  we output only entries for which the whole lineage fulfils the conditions at all levels tested

Provided by Jaimito

Ind ID	Archaeologist group	parentID
~~AR0001~~	~~Fulano~~	~~Site A~~
AR2222	Jaimito	Site G
~~AR5555~~	~~Jaimito~~	~~Site G~~
AR9999	Jaimito	Site Z

Labelled
before November 2021

Ske Ele ID	Date Labelling	parentID
AR0001.1	2021-04-12	AR0001.1
AR2222.1	2021-05-22	AR0001.1
~~AR5555.1~~	~~2021-05-22~~	~~AR5555~~
~~AR9999.1~~	~~2021-11-01~~	~~AR9999.1~~
AR9999.2	2021-04-12	AR9999.2

Root or Root Apex
More than 5 mg

Extract ID	Quantity	Extract Type	parentID
~~AR0001.1.01~~	~~10mg~~	~~Pulp~~	~~AR0001.1~~
AR0001.1.02	10mg	Root apex	AR0001.1
AR2222.1.01	100mg	Root	AR2222.1
~~AR2222.1.02~~	~~4mg~~	~~Root Apex~~	~~AR2222.1~~
AR9999.1.01	45mg	Root	AR9999.1
AR9999.2.01	30mg	Root	AR9999.2

API for downloading data applying filters

A bottom-up approach

- It is a AND logic function across Sample type levels:
  we output only entries for which the whole lineage fulfils the conditions at all levels tested

Site ID	Main Geographic region
~~Site A~~	~~La Pampa~~
~~Site B~~	~~Chubut~~
Site G	La Pampa
Site Z	Buenos Aires

Provided by Jaimito

Ind ID	Archaeologist group	parentID
~~AR0001~~	~~Fulano~~	~~Site A~~
AR2222	Jaimito	Site G
~~AR5555~~	~~Jaimito~~	~~Site G~~
AR9999	Jaimito	Site Z

Labelled
before November 2021

Ske Ele ID	Date Labelling	parentID
AR0001.1	2021-04-12	AR0001.1
AR2222.1	2021-05-22	AR0001.1
~~AR5555.1~~	~~2021-05-22~~	~~AR5555~~
~~AR9999.1~~	~~2021-11-01~~	~~AR9999.1~~
AR9999.2	2021-04-12	AR9999.2

Root or Root Apex
More than 5 mg

Extract ID	Quantity	Extract Type	parentID
~~AR0001.1.01~~	~~10mg~~	~~Pulp~~	~~AR0001.1~~
AR0001.1.02	10mg	Root apex	AR0001.1
AR2222.1.01	100mg	Root	AR2222.1
~~AR2222.1.02~~	~~4mg~~	~~Root Apex~~	~~AR2222.1~~
AR9999.1.01	45mg	Root	AR9999.1
AR9999.2.01	30mg	Root	AR9999.2

API for downloading data applying filters

A bottom-up approach

- It is a AND logic function across Sample type levels:
  we output only entries for which the whole lineage fulfils the conditions at all levels tested

From
La Pampa OR Chubut

Site ID	Main Geographic region
~~Site A~~	~~La Pampa~~
~~Site B~~	~~Chubut~~
Site G	La Pampa
~~Site Z~~	~~Buenos Aires~~

Provided by Jaimito

Ind ID	Archaeologist group	parentID
~~AR0001~~	~~Fulano~~	~~Site A~~
AR2222	Jaimito	Site G
~~AR5555~~	~~Jaimito~~	~~Site G~~
AR9999	Jaimito	Site Z

Labelled
before November 2021

Ske Ele ID	Date Labelling	parentID
AR0001.1	2021-04-12	AR0001.1
AR2222.1	2021-05-22	AR0001.1
~~AR5555.1~~	~~2021-05-22~~	~~AR5555~~
~~AR9999.1~~	~~2021-11-01~~	~~AR9999.1~~
AR9999.2	2021-04-12	AR9999.2

Root or Root Apex
More than 5 mg

Extract ID	Quantity	Extract Type	parentID
~~AR0001.1.01~~	~~10mg~~	~~Pulp~~	~~AR0001.1~~
AR0001.1.02	10mg	Root apex	AR0001.1
AR2222.1.01	100mg	Root	AR2222.1
~~AR2222.1.02~~	~~4mg~~	~~Root Apex~~	~~AR2222.1~~
AR9999.1.01	45mg	Root	AR9999.1
AR9999.2.01	30mg	Root	AR9999.2

API for downloading data applying filters

A bottom-up approach

- It is a AND logic function across Sample type levels:
  we output only entries for which the whole lineage fulfils the conditions at all levels tested

From
La Pampa OR Chubut

Site ID	Main Geographic region
~~Site A~~	~~La Pampa~~
~~Site B~~	~~Chubut~~
Site G	La Pampa
~~Site Z~~	~~Buenos Aires~~

Provided by Jaimito

Ind ID	Archaeologist group	parentID
~~AR0001~~	~~Fulano~~	~~Site A~~
AR2222	Jaimito	Site G
~~AR5555~~	~~Jaimito~~	~~Site G~~
~~AR9999~~	~~Jaimito~~	~~Site Z~~

Labelled
before November 2021

Ske Ele ID	Date Labelling	parentID
~~AR0001.1~~	~~2021-04-12~~	~~AR0001.1~~
AR2222.1	2021-05-22	AR0001.1
~~AR5555.1~~	~~2021-05-22~~	~~AR5555~~
~~AR9999.1~~	~~2021-11-01~~	~~AR9999.1~~
~~AR9999.2~~	~~2021-04-12~~	~~AR9999.2~~

Root or Root Apex
More than 5 mg

Extract ID	Quantity	Extract Type	parentID
~~AR0001.1.01~~	~~10mg~~	~~Pulp~~	~~AR0001.1~~
~~AR0001.1.02~~	~~10mg~~	~~Root apex~~	~~AR0001.1~~
AR2222.1.01	100mg	Root	AR2222.1
~~AR2222.1.02~~	~~4mg~~	~~Root Apex~~	~~AR2222.1~~
~~AR9999.1.01~~	~~45mg~~	~~Root~~	~~AR9999.1~~
~~AR9999.2.01~~	~~30mg~~	~~Root~~	~~AR9999.2~~

API for downloading data applying filters

Different filters according to feature type

- Date: filter IN samples with date
  within a period of time [Eldest; Most Recent]

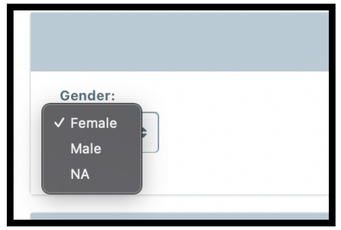
```
def getDateFilter():
    wrongEntry=True
    while wrongEntry:
        MostRecent="?"
        while MostRecent != "9999-12-31" and not CheckDate(MostRecent):
            MostRecent=input("Enter the most recent date, i.e. we will filter IN samples before that date (type
            if MostRecent == "Any":
                MostRecent="9999-12-31"
        MostRecent=datetime.strptime(MostRecent,'%Y-%m-%d')
        Eldest="?"
        while Eldest != "0001-01-01" and not CheckDate(Eldest):
            Eldest=input("Enter the eldest date, i.e. i.e. we will filter IN samples after that date (type Any
            if Eldest == "Any":
                Eldest="0001-01-01"
        Eldest=datetime.strptime(Eldest,'%Y-%m-%d')
        if Eldest<MostRecent:
            wrongEntry=False
        else:
            print("you entered a mostRecent date more ancient and EldestDate")

    return({"MostRecent":MostRecent,"Eldest":Eldest})
```

```
def filterDate(value,filter):
    value=datetime.strptime(value,'%Y-%m-%d')
    return(value<=filter["MostRecent"] and value>=filter["Eldest"])
```

API for downloading data applying filters

Different filters according to feature type

- Options: for Dropdown menus or checkboxes
  filter IN samples with one of the options entered




```python
def getOptionFilter(possibleChoices):
    print(len(possibleChoices))
    wrongEntry=True
    while wrongEntry:
        print("possible choices")
        index=0
        for value in possibleChoices:
            index=index+1
            print(format(index)+":"+value)
        listEntered=input("enter your choice(s) (the number(s) separated by space)").split()
        listEntered=[int(i)-1 for i in listEntered ]
        if min(listEntered) <0 or max(listEntered)>=len(possibleChoices):
            print("you entered choices out of range")
        else:
            wrongEntry=False
    return([possibleChoices[i] for i in listEntered])

```

```python
def filterCombo(value,filter):
    return(value in filter)
```

```python
def filterCheckbox(value,filter):
    AllFound=True
    for i in value:
        if i not in filter:
            AllFound=False
    return(AllFound)
```

API for downloading data applying filters

Different filters according to feature type

- Free Text: filter IN samples with a given string found in text (very basic….)

```
###for now we cover just the case where a given string is in the feature (no filter for NOT, OR, AND, NOT ANY,
def getTextFilter():
    return(input("enter a string to find in the field"))
```

```
def filterText(value,filter):
    return(filter in value)

```

API for downloading data applying filters

Different filters according to feature type

- By Sample Names: filter IN or OUT samples with ID or parentID in a given list of IDs

```python
import re

def getLinkFilter(sampleType,allIDs,link):
    parentPattern={"Site":{"pattern":"Any","typeParent":"None"},
                   "Individual":{"pattern":'[A][R][0-9][0-9][0-9][0-9]',"typeParent":"Site"},
                   "Skeleton Element":{"pattern":'[AR][0-9][0-9][0-9][0-9][.][0-9]',"typeParent":"Individual"},
                   "Extract":{"pattern":'[AR][0-9][0-9][0-9][0-9][.][0-9][.][0-9]',"typeParent":"Skeleton Eleme

    if sampleType not in parentPattern.keys():
        raise(sampleType+" not covered to retrieve its parent sample")
    if link:
        typeToCheck=parentPattern[sampleType][typeParent]
    else:
        typeToCheck=sampleType

    listType="?"
    while not listType in ["prompt","file"]:
        listType=input("will you enter IDs one by one or a file (prompt/file)?")
    wrongEntry=True
    while wrongEntry:
        if listType=="file":
            listType=open(input("file with parent file"),"r").readlines()
            listID=[]
            for i in listType:
                listID.append(i.strip())
        else:
            listID=input("enter the parent sample IDs separated by <space>/<space>, must match pattern "+parent
            listID=listID.split(" / ")
        wrongEntry=False
        for id in listID:
            ###check all id match pattern
            if not (re.match(parentPattern[typeToCheck]["pattern"],id) or parentPattern[typeToCheck]["pattern"]
                print("wrong pattern for "+id)
                wrongEntry=True
                ###check all id already registered
            if not id in allIDs.keys():
                print(id+" not registered in eLab")
                wrongEntry=True
        if wrongEntry:
            print("change those ids either in the file or in the prompted list")

    bound="?"
    while bound not in ["notin","in"]:
        bound=input("keep or remove those IDS (in/notin)?")
    return({"rule":bound,"list":listID})
```

```python
def filterName(value,listNAM,ruler):
    if ruler=="in":
        return(value in listNAM)
    elif ruler=="notin":
        return(value not in listNAM)
    else:
        raise()
```

```python
def filterLink(value,listNAM,ruler):
    value=value.split("|")[0]
    if ruler=="in":
        return(value in listNAM)
    elif ruler=="notin":
        return(value not in listNAM)
    else:
        raise()
```

API for downloading data applying filters

Different filters according to feature type

- By Quantity: filter IN samples with quantity >, < or = to a given quantity

```python
def getQuantityFilter():
    wrongEntry=True
    while wrongEntry:
        quanti=float(input("enter a quantity"))
        bound=input("enter a bound (less, more, exact)")
        if bound in ["less","more","exact"]:
            wrongEntry=False
    return({"rule":bound,"quantity":quanti})


```

```python
def filterQuantity(value,thres,ruler):
    if ruler == "exact":
        return(value==thres)
    elif ruler == "less":
        return(value<=thres)
    elif ruler == "more":
        return(value>=thres)
    else:
        raise(ruler+ " not recognized")
```

API for downloading data applying filters

A generic way to define filters and info to output at all levels

- Everything is asked by prompt…. Can be time consuming