DataCollectionTool

June 25, 2022

1 Data collection tool

Author: B209978

1.1 This tool is designed to allow input of data from selected hospitals for further analysis

[68]:
```
#Load pandas package and import data
import pandas as pd
testData=pd.read_csv("../Data/ae_attendances_test.csv")
# view
testData
```

[68]:

	index	period	org_code	attendances	breaches	admissions
0	2881	2016-07-01	RXK	14488	2128	3141
1	2896	2016-07-01	RNA	8947	596	2599
2	4258	2018-03-01	RXK	13805	3556	3429
3	4281	2018-03-01	RRK	9936	2154	3896
4	5043	2018-01-01	RLQ	4532	1263	1437
5	6471	2017-09-01	RWP	9817	2716	2921
6	7137	2017-07-01	RJC	5811	297	1617
7	7509	2017-06-01	RWP	10313	2824	3174
8	9577	2018-12-01	RXK	13604	4432	3744
9	10327	2018-10-01	RKB	12519	1937	4407
10	12530	2018-04-01	RL4	10709	1704	2544

1.2 View data types

[69]:
```
# assign data types for later referral
result = testData.dtypes
# view data types
print("Output:")
print(result)
```

```
Output:
index           int64
period         object
org_code       object
```

```
attendances     int64
breaches        int64
admissions      int64
dtype: object
```

1.2.1 Create empty data frame to input data

[288]:
```
# create dataframe
dfTofill = pd.DataFrame({'index': [0],# Integer
                    'period': [pd.Timestamp('20000101')], # Date
                    'org_code': ['NA'], # String
                    'attendances': [0], # Integer
                    'breaches': [0], # Integer
                    'admissions': [0], # Integer
                    'breach_performance': [0.0], # Float
                    'admission_rate': [0.0], #Float
                    'consent': [False]}) # Boolean

# view
dfTofill
# save once then comment out to prevent accidental overwrite
#dfTofill.to_csv('../Data/CollectedData.csv', index=False)
```

[288]:

	index	period	org_code	attendances	breaches	admissions	breach_performance	admission_rate	consent
0	0	2000-01-01	NA	0	0	0	0.0	0.0	False

1.2.2 Read in empty dataframe to begin collection

[267]:
```
# read in
CollectData=pd.read_csv("../Data/CollectedData.csv")
# view
CollectData
```

[267]:

	index	period	org_code	attendances	breaches	admissions
0	2881	2016-07-01	RXK	14488	2128	3141
1	8947	2016-07-01	RNA	8947	596	2599
2	4258	2018-03-01	RXK	13805	3556	3429
3	4281	2018-03-01	RRK	9936	2154	3896
4	5043	2018-01-01	RLQ	4532	1263	1437
5	6471	2017-09-01	RWP	9817	2716	2921
6	7137	2017-07-01	RJC	5811	297	1617
7	7509	2017-06-01	RWP	10313	2824	3174
8	9577	2018-12-01	RXK	13604	4432	3744
9	10327	2018-10-01	RKB	12519	1937	4407

	breach_performance	admission_rate	consent
0	0.853120	0.216800	True
1	0.933385	0.290488	True
2	0.742412	0.248388	True
3	0.783213	0.392110	True
4	0.721315	0.317079	True
5	0.723337	0.297545	True
6	0.948890	0.278265	True
7	0.726171	0.307767	True
8	0.674213	0.275213	True
9	0.845275	0.352025	True

1.2.3 Input index variable

[268]:
```
# view current row of data - Remember Python 0 indexing!
current_record = testData.head(n=int(input("Record number")))
current_record
```

```
Record number11
```

[268]:

	index	period	org_code	attendances	breaches	admissions
0	2881	2016-07-01	RXK	14488	2128	3141
1	2896	2016-07-01	RNA	8947	596	2599
2	4258	2018-03-01	RXK	13805	3556	3429
3	4281	2018-03-01	RRK	9936	2154	3896
4	5043	2018-01-01	RLQ	4532	1263	1437
5	6471	2017-09-01	RWP	9817	2716	2921
6	7137	2017-07-01	RJC	5811	297	1617
7	7509	2017-06-01	RWP	10313	2824	3174
8	9577	2018-12-01	RXK	13604	4432	3744
9	10327	2018-10-01	RKB	12519	1937	4407
10	12530	2018-04-01	RL4	10709	1704	2544

[269]:
```
# ask user for index number
index_number=input("Index number")
# use input to fill dataframe
dfTofill.iloc[0,0]=index_number
# view updated dataframe
dfTofill
```

```
Index number12530
```

[269]:

	index	period	org_code	attendances	breaches	admissions	\
0	12530	2018-10-01	RKB	12519	1937	4407	

	breach_performance	admission_rate	consent

```
0            0.845275        0.352025     True
```

1.3 Using widgets for data collection

[270]:
```
#Load the 'ipywidgets' package
import ipywidgets as widgets
#Load the 'IPython.display' package
from IPython.display import display
```

1.3.1 Consent checkbox

Ensure consent is gained and tick box to ensure collected data can be used

[271]:
```
# create boolean widget and default to false to ensure consent is gained
a = widgets.Checkbox(
    value=False,
    description='I consent for the data I have provided to be processed and␣
 ↪shared in accordance with data protection regulations with the purpose of␣
 ↪improving care service provision across the UK.',
    disabled=False
)
display(a)
```

```
Checkbox(value=False, description='I consent for the data I have provided to be processed and s
```

[272]:
```
# input consent response
dfTofill.iloc[0,8]=a.value
# view updated dataframe
dfTofill
```

[272]:
```
   index      period org_code  attendances  breaches  admissions  \
0  12530  2018-10-01      RKB        12519      1937        4407

   breach_performance  admission_rate  consent
0            0.845275        0.352025     True
```

1.3.2 Input period

[273]:
```
# view data type
print(result[1])
# view current record
current_record
```

```
object
```

[273]:

	index	period	org_code	attendances	breaches	admissions
0	2881	2016-07-01	RXK	14488	2128	3141
1	2896	2016-07-01	RNA	8947	596	2599
2	4258	2018-03-01	RXK	13805	3556	3429
3	4281	2018-03-01	RRK	9936	2154	3896
4	5043	2018-01-01	RLQ	4532	1263	1437
5	6471	2017-09-01	RWP	9817	2716	2921
6	7137	2017-07-01	RJC	5811	297	1617
7	7509	2017-06-01	RWP	10313	2824	3174
8	9577	2018-12-01	RXK	13604	4432	3744
9	10327	2018-10-01	RKB	12519	1937	4407
10	12530	2018-04-01	RL4	10709	1704	2544

[274]:
```
# create date picker widget
b = widgets.DatePicker(
    description='Period',
    disabled=False
)
display(b)
```

```
DatePicker(value=None, description='Period')
```

[275]:
```
# input period response
dfTofill.iloc[0,1]=b.value
# view updated dataframe
dfTofill
```

[275]:

	index	period	org_code	attendances	breaches	admissions	\
0	12530	2018-04-01	RKB	12519	1937	4407	

	breach_performance	admission_rate	consent
0	0.845275	0.352025	True

1.3.3 Input organisation code

[276]:
```
# view data type
print(result[2])
# create list of unique organisation codes
org_code=list(testData['org_code'].unique())
# view list
print(org_code)
# view current record
current_record
```

```
object
['RXK', 'RNA', 'RRK', 'RLQ', 'RWP', 'RJC', 'RKB', 'RL4']
```

[276]:

	index	period	org_code	attendances	breaches	admissions
0	2881	2016-07-01	RXK	14488	2128	3141
1	2896	2016-07-01	RNA	8947	596	2599
2	4258	2018-03-01	RXK	13805	3556	3429
3	4281	2018-03-01	RRK	9936	2154	3896
4	5043	2018-01-01	RLQ	4532	1263	1437
5	6471	2017-09-01	RWP	9817	2716	2921
6	7137	2017-07-01	RJC	5811	297	1617
7	7509	2017-06-01	RWP	10313	2824	3174
8	9577	2018-12-01	RXK	13604	4432	3744
9	10327	2018-10-01	RKB	12519	1937	4407
10	12530	2018-04-01	RL4	10709	1704	2544

[43]:
```
# create selection widget
c=widgets.Select(
    options=org_code,
    value='RXK',
    rows=len(org_code),
    description='ODS code:',
    disabled=False
)
display(c)
```

```
Select(description='ODS code:', options=('RXK', 'RNA', 'RRK', 'RLQ', 'RWP', 'RJC', 'RKB', 'RL4
```

[277]:
```
# input org_code response
dfTofill.iloc[0,2]=c.value
# view updated dataframe
dfTofill
```

[277]:

	index	period	org_code	attendances	breaches	admissions	\
0	12530	2018-04-01	RL4	12519	1937	4407	

	breach_performance	admission_rate	consent
0	0.845275	0.352025	True

1.3.4 Input attendances

[278]:
```
# view data type
print(result[3])
# view current record
current_record
```

```
int64
```

[278]:

	index	period	org_code	attendances	breaches	admissions
0	2881	2016-07-01	RXK	14488	2128	3141
1	2896	2016-07-01	RNA	8947	596	2599
2	4258	2018-03-01	RXK	13805	3556	3429
3	4281	2018-03-01	RRK	9936	2154	3896
4	5043	2018-01-01	RLQ	4532	1263	1437
5	6471	2017-09-01	RWP	9817	2716	2921
6	7137	2017-07-01	RJC	5811	297	1617
7	7509	2017-06-01	RWP	10313	2824	3174
8	9577	2018-12-01	RXK	13604	4432	3744
9	10327	2018-10-01	RKB	12519	1937	4407
10	12530	2018-04-01	RL4	10709	1704	2544

[258]:
```
# create integer widget
e=widgets.IntText(
    value=0,
    description='Attendances:',
    disabled=False)
display(e)
```

```
IntText(value=0, description='Attendances:')
```

[279]:
```
# input attendance response
dfTofill.iloc[0,3]=e.value
# view updated dataframe
dfTofill
```

[279]:

	index	period	org_code	attendances	breaches	admissions	\
0	12530	2018-04-01	RL4	10709	1937	4407	

	breach_performance	admission_rate	consent
0	0.845275	0.352025	True

1.3.5 Define function to calculate breach perfomance and admission rate

[280]:
```
def metrics(column):
    return dfTofill.iloc[0,column]/dfTofill.iloc[0,3]
```

1.3.6 Input breaches

Breach_performance automatically calculated using 1-breaches/attendances

[281]:
```
# view data type
print(result[4])
# view current record
current_record
```

```
int64
```

[281]:

	index	period	org_code	attendances	breaches	admissions
0	2881	2016-07-01	RXK	14488	2128	3141
1	2896	2016-07-01	RNA	8947	596	2599
2	4258	2018-03-01	RXK	13805	3556	3429
3	4281	2018-03-01	RRK	9936	2154	3896
4	5043	2018-01-01	RLQ	4532	1263	1437
5	6471	2017-09-01	RWP	9817	2716	2921
6	7137	2017-07-01	RJC	5811	297	1617
7	7509	2017-06-01	RWP	10313	2824	3174
8	9577	2018-12-01	RXK	13604	4432	3744
9	10327	2018-10-01	RKB	12519	1937	4407
10	12530	2018-04-01	RL4	10709	1704	2544

[51]:
```
# create integer widget
f=widgets.IntText(
    value=0,
    description='Breaches:',
    disabled=False)
display(f)
```

```
IntText(value=0, description='Breaches:')
```

[282]:
```
# input breaches response
dfTofill.iloc[0,4]=f.value
# calculate breach_performance
g=1-(metrics(4))
# input calculated breach_performance
dfTofill.iloc[0,6]=g
# view updated dataframe
dfTofill
```

[282]:

	index	period	org_code	attendances	breaches	admissions	breach_performance	admission_rate	consent
0	12530	2018-04-01	RL4	10709	1704	4407	0.840882	0.352025	True

1.3.7 Input admissions

Addmission_rate automatically calculated using admissions/attendances

[283]:
```
# view data type
print(result[5])
# view current record
current_record
```

```
int64
```

[283]:

	index	period	org_code	attendances	breaches	admissions
0	2881	2016-07-01	RXK	14488	2128	3141
1	2896	2016-07-01	RNA	8947	596	2599
2	4258	2018-03-01	RXK	13805	3556	3429
3	4281	2018-03-01	RRK	9936	2154	3896
4	5043	2018-01-01	RLQ	4532	1263	1437
5	6471	2017-09-01	RWP	9817	2716	2921
6	7137	2017-07-01	RJC	5811	297	1617
7	7509	2017-06-01	RWP	10313	2824	3174
8	9577	2018-12-01	RXK	13604	4432	3744
9	10327	2018-10-01	RKB	12519	1937	4407
10	12530	2018-04-01	RL4	10709	1704	2544

[64]:
```
# create integer widget
h=widgets.IntText(
    value=0,
    description='Admissions:',
    disabled=False)
display(h)
```

```
IntText(value=0, description='Admissions:')
```

[284]:
```
# input admissions response
dfTofill.iloc[0,5]=h.value
# calculate admission_rate
i=(metrics(5))
# input calculated breach_performance
dfTofill.iloc[0,7]=i
# view update dataframe
dfTofill
```

[284]:

	index	period	org_code	attendances	breaches	admissions	\
0	12530	2018-04-01	RL4	10709	1704	2544	

	breach_performance	admission_rate	consent
0	0.840882	0.237557	True

2 Concatenating the collected data to the CollectData data frame.

Let us use the `concat()` function from the Python *pandas* package to append the CollectData and dfTofill data frames. The concat() function is used to concatenate *pandas* objects.

[285]:
```
# add completed row to dataframe
CollectData  = pd.concat([CollectData, dfTofill])
```

```
# view updated datafrme
display(CollectData)
```

```
   index      period org_code  attendances  breaches  admissions  \
0   2881  2016-07-01      RXK        14488      2128        3141
1   8947  2016-07-01      RNA         8947       596        2599
2   4258  2018-03-01      RXK        13805      3556        3429
3   4281  2018-03-01      RRK         9936      2154        3896
4   5043  2018-01-01      RLQ         4532      1263        1437
5   6471  2017-09-01      RWP         9817      2716        2921
6   7137  2017-07-01      RJC         5811       297        1617
7   7509  2017-06-01      RWP        10313      2824        3174
8   9577  2018-12-01      RXK        13604      4432        3744
9  10327  2018-10-01      RKB        12519      1937        4407
0  12530  2018-04-01      RL4        10709      1704        2544

   breach_performance  admission_rate  consent
0            0.853120        0.216800     True
1            0.933385        0.290488     True
2            0.742412        0.248388     True
3            0.783213        0.392110     True
4            0.721315        0.317079     True
5            0.723337        0.297545     True
6            0.948890        0.278265     True
7            0.726171        0.307767     True
8            0.674213        0.275213     True
9            0.845275        0.352025     True
0            0.840882        0.237557     True
```

2.0.1 Only retain responses where consent has been received

[286]:
```
# filter out false consent responses
CollectData=CollectData[CollectData['consent'] == True]
# view updaed dataframe
display(CollectData)
# save collected data for next response
CollectData.to_csv('../Data/CollectedData.csv', index=False)
```

```
   index      period org_code  attendances  breaches  admissions  \
0   2881  2016-07-01      RXK        14488      2128        3141
1   8947  2016-07-01      RNA         8947       596        2599
2   4258  2018-03-01      RXK        13805      3556        3429
3   4281  2018-03-01      RRK         9936      2154        3896
4   5043  2018-01-01      RLQ         4532      1263        1437
5   6471  2017-09-01      RWP         9817      2716        2921
6   7137  2017-07-01      RJC         5811       297        1617
```

```
7   7509  2017-06-01      RWP        10313      2824        3174
8   9577  2018-12-01      RXK        13604      4432        3744
9  10327  2018-10-01      RKB        12519      1937        4407
0  12530  2018-04-01      RL4        10709      1704        2544

   breach_performance  admission_rate  consent
0            0.853120        0.216800     True
1            0.933385        0.290488     True
2            0.742412        0.248388     True
3            0.783213        0.392110     True
4            0.721315        0.317079     True
5            0.723337        0.297545     True
6            0.948890        0.278265     True
7            0.726171        0.307767     True
8            0.674213        0.275213     True
9            0.845275        0.352025     True
0            0.840882        0.237557     True
```

2.0.2 Save completed data collection tool

[287]:
```
CollectData.to_csv('../RawData/CollectedDataFinal.csv', index=False)
```