data_collection_tool

June 20, 2022

1 Title: Collecting data using interactive Jupyter widgets

Author details: *Author:* Shona McElroy. *Contact details:* s2272790@ed.ac.uk. **Notebook and data info:** This Notebook provides for part of the assessment for Working with Data Types and Structures in R and Python using interactive jupyter-widgets and to collect the NHS England mortality data (ons_mortality).The following widgets are designed to capture the data required for a data capture tool. **Data:** Data consists of date, numerical data and character data from NHSRdatasets package. **Copyright statement:** This Notebook is the product of Shona McElroy.

[1]:
```
#Load the 'pandas' package
import pandas as pd
testData=pd.read_csv("../Data/ons_mortality_ENG_1019_test.csv")
testData
```

[1]:

	index	week_no	year	date	counts	mort_avg	variance_from_avg
0	1798	30	2011	2011-07-29	8456	8737.4	0.967794
1	1803	35	2011	2011-09-02	7717	7984.6	0.966485
2	5324	19	2013	2013-05-10	8814	9096.0	0.968997
3	5326	21	2013	2013-05-24	9530	9311.2	1.023499
4	7121	48	2014	2014-11-28	9928	9398.8	1.056305
5	10722	27	2016	2016-07-08	9138	8872.2	1.029959
6	10742	47	2016	2016-11-25	10603	9572.0	1.107710
7	12473	10	2017	2017-03-10	11077	10816.0	1.024131
8	14259	28	2018	2018-07-13	9293	9018.0	1.030495
9	16021	22	2019	2019-05-31	8260	8125.0	1.016615
10	16029	30	2019	2019-07-26	9112	9023.0	1.009864

Data type

[2]:
```
result = testData.dtypes
print("Output:")
print(result)
```

```
Output:
index                 int64
week_no               int64
year                  int64
date                 object
counts                int64
```

```
mort_avg             float64
variance_from_avg    float64
dtype: object
```

View a sample of the test data frame

[3]:
```
testData.head(n=1)
```

[3]:

	index	week_no	year	date	counts	mort_avg	variance_from_avg
0	1798	30	2011	2011-07-29	8456	8737.4	0.967794

Set up empty data frame for data collection

[4]:
```
dfTofill = pd.DataFrame({'index': [0],# Integer
                    'week_no': [0], # Integer
                    'year': [0], # Integer
                    'date': [pd.Timestamp('20000101')], # Date
                    'counts': [0], # Integer
                    'mort_avg': [0.0], # Float
                    'variance_from_avg': [0], # Integer
                    'consent': [False]}) # Boolean

dfTofill
```

[4]:

	index	week_no	year	date	counts	mort_avg	variance_from_avg	consent
0	0	0	0	2000-01-01	0	0.0	0	False

Save the empty data frame

[252]:
```
#dfTofill.to_csv('../Data/collected_data.csv', index=False)
```

[5]:
```
CollectData=pd.read_csv("../Data/collected_data.csv")
CollectData
```

[5]:

	index	week_no	year	date	counts	mort_avg	variance_from_avg
0	1798	30	2011	2011-07-29	0	8737.4	0.967794
1	1803	35	2011	2011-09-02	7717	7984.6	0.966485
2	5324	19	2013	2013-05-10	8814	9096.0	0.968997
3	5326	21	2013	2013-05-24	9530	9311.2	1.023499
4	7121	48	2014	2014-11-28	9928	9398.8	1.056305
5	10722	27	2016	2016-07-08	9138	8872.2	1.029959
6	10742	47	2016	2016-11-25	10603	9572.0	1.107710
7	12473	10	2017	2017-03-10	11077	10816.0	1.024131
8	14259	28	2018	2018-07-13	9293	9018.0	1.030495
9	16021	22	2019	2019-05-31	8260	8125.0	1.016615

```
10  16029       30  2019  2019-07-26    9112    9023.0           1.009864

    consent
0      True
1      True
2      True
3      True
4      True
5      True
6      True
7      True
8      True
9      True
10     True
```

1.1 Index number for each record (to be changed for each entry)

[255]:
```
index_number=16029 #Remember to change for each record.
```

[256]:
```
dfTofill.iloc[0,0]=index_number
dfTofill
```

[256]:
```
   index  week_no  year       date  counts  mort_avg  variance_from_avg  \
0  16029        0     0 2000-01-01       0       0.0                  0

   consent
0    False
```

Load the widgets and display packages

[6]:
```
#Load the 'ipywidgets' package
import ipywidgets as widgets
#Load the 'IPython.display' package
from IPython.display import display
```

1.2 Week Number

[7]:
```
b = widgets.BoundedIntText(
    value=1,
    min=1,
    max=52,
    step=1,
    description='Week Number:',
    style={'description_width': 'initial'},
    layout={'width': 'max-content'},
    disabled=False
)
```

```
display(b)
```

```
BoundedIntText(value=1, description='Week Number:', layout=Layout(width='max-content'), max=52
```

[8]:
```
dfTofill.iloc[0,1]=b.value
dfTofill
```

[8]:
```
   index  week_no  year       date  counts  mort_avg  variance_from_avg  \
0      0       30     0 2000-01-01       0       0.0                  0

   consent
0    False
```

1.3 Year

[9]:
```
c = widgets.Dropdown(
    options=['2010', '2011', '2012', '2013', '2014', '2015', '2016', '2017',␣
 ↪'2018', '2019', '2020', '2021', '2022', '2023', '2024'],
    value='2022',
    description='Number:',
    disabled=False,
)
display(c)
```

```
Dropdown(description='Number:', index=12, options=('2010', '2011', '2012', '2013', '2014', '201
```

[10]:
```
dfTofill.iloc[0,2]=c.value
dfTofill
```

[10]:
```
   index  week_no  year       date  counts  mort_avg  variance_from_avg  \
0      0       30  2019 2000-01-01       0       0.0                  0

   consent
0    False
```

1.4 Date

[11]:
```
d = widgets.DatePicker(
    description='Period',
    disabled=False
)
display(d)
```

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

[12]:
```
dfTofill.iloc[0,3]=d.value
dfTofill
```

[12]:

	index	week_no	year	date	counts	mort_avg	variance_from_avg	consent
0	0	30	2019	2019-07-27	0	0.0	0	False

1.5 Counts

[13]:
```
e = widgets.IntText(
        value=0,
        description='Number of deaths in the preceding week:',
        disabled=False,
        style={'description_width': 'initial'},
        layout={'width': 'max-content'})
display(e)
```

```
IntText(value=0, description='Number of deaths in the preceding week:', layout=Layout(width='ma
```

[14]:
```
dfTofill.iloc[0,4]=e.value
dfTofill
```

[14]:

	index	week_no	year	date	counts	mort_avg	variance_from_avg	consent
0	0	30	2019	2019-07-27	9112	0.0	0	False

1.6 Average mortality

[15]:
```
f = widgets.FloatText(
        value=0.0,
        description='Average deaths for this week in the preceding 5-years:',
        disabled=False,
        style={'description_width': 'initial'},
        layout={'width': 'max-content'})
display(f)
```

```
FloatText(value=0.0, description='Average deaths for this week in the preceding 5-years:', lay
```

[17]:
```
dfTofill.iloc[0,5]=f.value
dfTofill
```

[17]:

	index	week_no	year	date	counts	mort_avg	variance_from_avg	consent
0	0	30	2019	2019-07-27	9112	9023.0	0	False

1.7 Variance from Average

[18]:
```
g=widgets.FloatText(
    value=0.0,
    description='Average deaths for this week (5-years preceding):',
    disabled=False,
    style={'description_width': 'initial'},
    layout={'width': 'max-content'}
)
display(g)
```

```
FloatText(value=0.0, description='Average deaths for this week (5-years preceding):', layout=La
```

[19]:
```
dfTofill.iloc[0,6]=g.value
dfTofill
```

[19]:

	index	week_no	year	date	counts	mort_avg	variance_from_avg	consent
0	0	30	2019	2019-07-27	9112	9023.0	1.009864	False

1.8 Consent

[20]:
```
h = 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,
    style={'description_width': 'initial'},
    layout={'width': 'max-content'}
)
display(h)
```

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

[21]:
```
dfTofill.iloc[0,7]=h.value
dfTofill
```

[21]:

	index	week_no	year	date	counts	mort_avg	variance_from_avg	consent
0	0	30	2019	2019-07-27	9112	9023.0	1.009864	True

2 Concatenate 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.

[271]:
```
# CollectData is the first data frame
# dfTofill is the second data frame
CollectData  = pd.concat([CollectData, dfTofill])
display(CollectData)
```

	index	week_no	year	date	counts	mort_avg	variance_from_avg	consent
0	1798	30	2011	2011-07-29	0	8737.4	0.967794	True
1	1803	35	2011	2011-09-02	7717	7984.6	0.966485	True
2	5324	19	2013	2013-05-10	8814	9096.0	0.968997	True
3	5326	21	2013	2013-05-24	9530	9311.2	1.023499	True
4	7121	48	2014	2014-11-28	9928	9398.8	1.056305	True
5	10722	27	2016	2016-07-08	9138	8872.2	1.029959	True
6	10742	47	2016	2016-11-25	10603	9572.0	1.107710	True
7	12473	10	2017	2017-03-10	11077	10816.0	1.024131	True
8	14259	28	2018	2018-07-13	9293	9018.0	1.030495	True
9	16021	22	2019	2019-05-31	8260	8125.0	1.016615	True
0	16029	30	2019	2019-07-26	9112	9023.0	1.009864	True

2.0.1 Check consent has been given

[272]:
```
CollectData=CollectData[CollectData['consent'] == True]
display(CollectData)
```

```
   index  week_no  year        date  counts  mort_avg  variance_from_avg  \
0   1798       30  2011  2011-07-29       0    8737.4           0.967794
1   1803       35  2011  2011-09-02    7717    7984.6           0.966485
2   5324       19  2013  2013-05-10    8814    9096.0           0.968997
3   5326       21  2013  2013-05-24    9530    9311.2           1.023499
4   7121       48  2014  2014-11-28    9928    9398.8           1.056305
5  10722       27  2016  2016-07-08    9138    8872.2           1.029959
6  10742       47  2016  2016-11-25   10603    9572.0           1.107710
7  12473       10  2017  2017-03-10   11077   10816.0           1.024131
8  14259       28  2018  2018-07-13    9293    9018.0           1.030495
9  16021       22  2019  2019-05-31    8260    8125.0           1.016615
0  16029       30  2019  2019-07-26    9112    9023.0           1.009864

   consent
0     True
1     True
2     True
3     True
4     True
5     True
6     True
7     True
8     True
9     True
0     True
```

2.0.2 Save the CollectData data frame

[273]:
```
CollectData.to_csv('../Data/collected_data.csv', index=False)
```

2.0.3 Save the completed CollectData file to RawData

[274]:
```
CollectData.to_csv('../RawData/collected_data_final.csv', index=False)
```

[ ]:
```

```