The Primary School - Assessment Task 2

In [1]:
```
import pandas as pd
import numpy as np
```

In [2]:
```
df = pd.read_csv('data_manager_work_sample.csv')
```

In [3]:
```
df.shape
```

Out[3]:
```
(4811, 15)
```

In [4]:
```
df.columns
```

Out[4]:
```
Index(['id', 'iep', 'ell', 'grade', 'expected_small', 'expected_1', 'week',
       'days_absent', 'whole_group', 'small_group', 'one_one', 'num_complete',
       'percent_complete', 'lexia_min', 'dreambox_min'],
      dtype='object')
```

In [5]:
```
df.head(3)
```

Out[5]:

	id	iep	ell	grade	expected_small	expected_1	week	days_absent	whole_group	sı
0	705.0	N	NaN	PS	NaN	NaN	2020-08-19	0.0	NaN	
1	705.0	N	NaN	PS	NaN	NaN	2020-08-24	0.0	NaN	
2	705.0	N	NaN	PS	NaN	NaN	2020-08-31	0.0	NaN	

1. How many unique students are represented in the dataset?

In [6]:
```
# Inclues null entries
df['id'].nunique(dropna=False)
```

Out[6]:
```
253
```

In [7]:
```
# There are 26 student rows/entries that have null ids
sum(df['id'].isnull())
```

Out[7]:
```
26
```

In [8]:
```
# Getting rid of null entries
a_unique = df[~df['id'].isnull()]['id'].nunique()
print('There are {} unique students'.format(a_unique))
# or
b_unique = df['id'].nunique(dropna=True)
print('There are {} unique students'.format(b_unique))
```

```
There are 252 unique students
There are 252 unique students
```

2. Which students and weeks had perfect attendance (0 absences)?

In [9]:
```
df2 = df[['id','week','days_absent']]
# Not perfect attendance
absence = df2['days_absent'].isin([1.0,2.0,3.0,4.0,np.nan]).sum()
print('Rows that have no perfect attendance: {}'.format(absence))
```

```
Rows that have no perfect attendance: 1540
```

Student IDS with perfect attendance

In [10]:
```
# List of student ids that have been absent at least once. They need to be fi
absent_ids = df2[df2['days_absent'].isin([1.0,2.0,3.0,4.0,np.nan])]['id']
# example
print(absent_ids[:3])
```

```
14    705.0
17    705.0
30    710.0
Name: id, dtype: float64
```

In [11]:
```
# Take all data points, then substract the students that have had at least on
ids_never_absent = df[~df['id'].isin(list(absent_ids))]['id'].unique()
print('These student ids have never been absent: {}'.format(ids_never_absent)
```

```
These student ids have never been absent: [700. 720. 742.]
```

In [15]:
```
# These are the weeks when students (700, 720, 742) attended.
df[~df['id'].isin(list(absent_ids))]['week'].unique()
```

Out[15]:
```
array(['2020-08-19', '2020-08-24', '2020-08-31', '2020-09-07',
       '2020-09-14', '2020-09-21', '2020-09-28', '2020-10-05',
       '2020-10-12', '2020-10-19', '2020-10-26', '2020-11-02',
       '2020-11-09', '2020-11-16', '2020-11-23', '2020-11-30',
       '2020-12-07', '2020-12-14'], dtype=object)
```

- **These student IDS have perfect attendance (0 absences): 700, 720, 742**

Weeks that had perfect attendance

In [16]:
```
# List of weeks that had at least one absence. They need to be filtered out.
absent_weeks_withnull = df2[df2['days_absent'].isin([1.0,2.0,3.0,4.0])]['week
absent_weeks_nonull = df2[df2['days_absent'].isin([1.0,2.0,3.0,4.0, np.nan])]
# example
print(absent_weeks_withnull[:3])
print(absent_weeks_nonull[:3])
```

```
14    2020-11-23
17    2020-12-14
30    2020-11-09
Name: week, dtype: object
14    2020-11-23
17    2020-12-14
30    2020-11-09
Name: week, dtype: object
```

In [17]:
```
weeks_no_absence_withnull = df[~df['week'].isin(list(absent_weeks_withnull))]
weeks_no_absence_nonull = df[~df['week'].isin(list(absent_weeks_nonull))]['we
print('These weeks had no absence (except null): {}'.format(weeks_no_absence_
print('These weeks had no absence (without null absences): {}'.format(weeks_n
```

```
These weeks had no absence (except null): ['2020-10-08']
These weeks had no absence (without null absences): []
```

In [18]:
```
# Further obserce
df[df['week'] == '2020-10-08']['days_absent'].unique()
# All 'days_absent' values on week 2020-10-08 are null
```

Out[18]:
```
array([nan])
```

- **There are overall no weeks which had perfect attendance, unless there was a null. (2020-10-08)**

3. Which grade is spending the most time on Lexia?

In [19]:
```
df.groupby('grade', dropna=False).sum()['lexia_min']
```

Out[19]:
```
grade
1st     3953.0
2nd     2879.0
3rd     2721.0
K       1200.0
PK         0.0
PS         0.0
NaN     1507.0
Name: lexia_min, dtype: float64
```

- **1st grade is spending the most on Lexia $3,953.**
- There is also 1,507 being allocated in Lexia but grade is not specified.

4. Which IEP students have had less than 10 1:1 meetings this term?

In [20]:
```
is_iep = df[df['iep']=='Y']
print('{} students are IEP'.format(is_iep['id'].nunique()))
```

```
57 students are IEP
```

In [21]:
```
# Mask dataframe that accounts for student ids that had less than 10 one on o
is_iep_less10_oneone = pd.DataFrame(is_iep.groupby('id').sum()['one_one']<10)
```

In [22]:
```
# Apply mask (ids of students will less than 10 one_one). Show student_id.
result = is_iep_less10_oneone[is_iep_less10_oneone['one_one']]['id']
print(result)
print('\n ids: {}'.format(list(result)))
```

```
26    504.0
27    513.0
28    521.0
29    549.0
30    586.0
32    679.0
33    704.0
34    710.0
35    715.0
36    718.0
37    732.0
38    734.0
39    739.0
40    789.0
Name: id, dtype: float64

 ids: [504.0, 513.0, 521.0, 549.0, 586.0, 679.0, 704.0, 710.0, 715.0, 718.0, 7
32.0, 734.0, 739.0, 789.0]
```

- These student ids had less than 10 one on one meetings during this term.

5. What percentage of students had 0 absences for all

weeks included in the file?

In [23]:
```
c = df['days_absent'].value_counts(dropna=False)
p = round(df['days_absent'].value_counts(dropna=False, normalize=True)*100, 2
pd.concat([c,p], axis=1, keys=['counts', '%'])
```

Out[23]:

	counts	%
0.0	3271	67.99
NaN	728	15.13
1.0	517	10.75
2.0	165	3.43
3.0	76	1.58
4.0	54	1.12

Students have been absent 0, 1, 2, 3, and 4 times.

- 15% of the entries in attendance are null.
- Most students have attended class 68% of the time

Final Notes

1. The database for this term contains too many null values in every single column. The most important one (id) also contains null values, which will create a lot of issues because we will be unable to identify which student it is referring to.
2. The column labels are not clear. We need to name the labels clearly, or provide adittional information (documentation).