캐글 데이터 분석

- 대회 URL : https://www.kaggle.com/c/learnplatform-covid19-impact-on-digital-learning/overview (https://www.kaggle.com/c/learnplatform-covid19-impact-on-digital-learning/overview)
- 평가 : https://www.kaggle.com/c/learnplatform-covid19-impact-on-digital-learning/overview/evaluation (https://www.kaggle.com/c/learnplatform-covid19-impact-on-digital-learning/overview/evaluation)
  - 명확성(Clarity)(5점)
  - 정확도(5점)
  - 창의성(5점)
- Timeline :
  - 2021/09/30 : 마지막 제출
  - 2021/10/28 : 수상자 발표

- 참고 노트북 : https://www.kaggle.com/iamleonie/gentle-introduction-to-the-dataset (https://www.kaggle.com/iamleonie/gentle-introduction-to-the-dataset)

In [3]:

```
import os
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

import re
import plotly.express as px
import plotly.graph_objects as go
from plotly.subplots import make_subplots
```

데이터 불러오기

In [36]:

```
path = os.getcwd()
print(path)
```

```
C:₩Users₩toto₩Documents₩Github₩KaggleDataAnalysis₩kaggle_1
```

In [37]:

```
districts_info = pd.read_csv("../../data/learnplatform-covid19-impact/districts_info.csv")
products_info = pd.read_csv("../../data/learnplatform-covid19-impact/products_info.csv")

districts_info.shape, products_info.shape
```

Out[37]:

```
((233, 7), (372, 6))
```

- districts_info : 각 학군에 대한 정보
- products_info : 디지털 학습에 사용되는 상위 370개 도구에 대한 정보 포함.

In [38]:

```
# engagement_data 폴더의 파일 확인
os.listdir("../../data/learnplatform-covid19-impact/engagement_data")
```

```
 '7798.csv',
 '7829.csv',
 '7858.csv',
 '7964.csv',
 '7970.csv',
 '7975.csv',
 '7980.csv',
 '8017.csv',
 '8076.csv',
 '8103.csv',
 '8127.csv',
 '8160.csv',
 '8184.csv',
 '8256.csv',
 '8328.csv',
 '8425.csv',
 '8433.csv',
 '8515.csv',
 '8520.csv',
 '8539.csv',
```

학군 정보

In [39]:

```
districts_info.head()
```

Out[39]:

	district_id	state	locale	pct_black/hispanic	pct_free/reduced	county_connections_ratio	pp
0	8815	Illinois	Suburb	[0, 0.2[	[0, 0.2[	[0.18, 1[	
1	2685	NaN	NaN	NaN	NaN	NaN	
2	4921	Utah	Suburb	[0, 0.2[	[0.2, 0.4[	[0.18, 1[	[6
3	3188	NaN	NaN	NaN	NaN	NaN	
4	2238	NaN	NaN	NaN	NaN	NaN	

370개의 학습 도구

In [40]:

```
products_info.head()
```

Out[40]:

	LP ID	URL	Product Name	Provider/Company Name	Sector(s)	Primary Essential Function
0	13117	https://www.splashmath.com	SplashLearn	StudyPad Inc.	PreK-12	LC - Digital Learning Platforms
1	66933	https://abcmouse.com	ABCmouse.com	Age of Learning, Inc	PreK-12	LC - Digital Learning Platforms
2	50479	https://www.abcya.com	ABCya!	ABCya.com, LLC	PreK-12	LC - Sites, Resources & Reference - Games & Si...
3	92993	http://www.aleks.com/	ALEKS	McGraw-Hill PreK-12	PreK-12; Higher Ed	LC - Digital Learning Platforms
4	73104	https://www.achieve3000.com/	Achieve3000	Achieve3000	PreK-12	LC - Digital Learning Platforms

In [41]:

```
# engagement_data 폴더의 파일 확인
list1 = os.listdir("../../data/learnplatform-covid19-impact/engagement_data")
list1[0:10]
```

Out[41]:

```
['1000.csv',
 '1039.csv',
 '1044.csv',
 '1052.csv',
 '1131.csv',
 '1142.csv',
 '1179.csv',
 '1204.csv',
 '1270.csv',
 '1324.csv']
```

- 폴더 안의 파일명은 [district_id].csv이 됩니다.

In [42]:

```
dis_info_1000 = pd.read_csv("../../data/learnplatform-covid19-impact/engagement_data/1000.csv")
dis_info_1000.head()
```

Out[42]:

	time	lp_id	pct_access	engagement_index
0	2020-01-01	93690.0	0.00	NaN
1	2020-01-01	17941.0	0.03	0.90
2	2020-01-01	65358.0	0.03	1.20
3	2020-01-01	98265.0	0.57	37.79
4	2020-01-01	59257.0	0.00	NaN

위의 파일들은 district_id와 lp_id로 결합이 가능하다.

02 데이터 전처리

- 모든 분석 대회에서 매우 중요한 단계
- 데이터 전처리를 시작하기에 앞서, 유지 및 수정하려는 데이터와 분석과 관련이 없는 데이터 생각해 보기

데이터 전처리 순서

- districts_info.csv : NaN states의 57개 학군 삭제
- products_info.csv : Sector(s)의 One-Hot Encode(원핫 인코딩)수행
- products_info.csv : 'Primary Essential Function' 컬럼을 기본 및 하위 범주로 분할

Dropping Districts with NaN States

In [43]:

```
print(districts_info.shape)
districts_info = districts_info[districts_info.state.notna()].reset_index(drop=True)
print(districts_info.shape)
```

```
(233, 7)
(176, 7)
```

One-Hot Encoding the Product Sectors

In [44]:

```
products_info['Sector(s)'].unique()
```

Out[44]:

```
array(['PreK-12', 'PreK-12; Higher Ed', 'PreK-12; Higher Ed; Corporate',
       nan, 'Corporate', 'Higher Ed; Corporate'], dtype=object)
```

In [45]:

```
temp_sectors = products_info['Sector(s)'].str.get_dummies(sep="; ")
temp_sectors.head()
```

Out[45]:

	Corporate	Higher Ed	PreK-12
0	0	0	1
1	0	0	1
2	0	0	1
3	0	1	1
4	0	0	1

In [46]:

```
temp_sectors.columns = [f"sector_{re.sub(' ', '', c)}" for c in temp_sectors.columns]
```

In [47]:

```
temp_sectors.columns
```

Out[47]:

```
Index(['sector_Corporate', 'sector_HigherEd', 'sector_PreK-12'], dtype='object')
```

In [48]:

```
products_info = products_info.join(temp_sectors)
products_info.head()
```

Out[48]:

	LP ID	URL	Product Name	Provider/Company Name	Sector(s)	Primary Essential Function
0	13117	https://www.splashmath.com	SplashLearn	StudyPad Inc.	PreK-12	LC - Digital Learning Platforms
1	66933	https://abcmouse.com	ABCmouse.com	Age of Learning, Inc	PreK-12	LC - Digital Learning Platforms
2	50479	https://www.abcya.com	ABCya!	ABCya.com, LLC	PreK-12	LC - Sites, Resources & Reference - Games & Si...
3	92993	http://www.aleks.com/	ALEKS	McGraw-Hill PreK-12	PreK-12; Higher Ed	LC - Digital Learning Platforms
4	73104	https://www.achieve3000.com/	Achieve3000	Achieve3000	PreK-12	LC - Digital Learning Platforms

In [49]:

```
products_info.drop("Sector(s)", axis=1, inplace=True)
print(products_info.columns)
del temp_sectors
```

```
Index(['LP ID', 'URL', 'Product Name', 'Provider/Company Name',
       'Primary Essential Function', 'sector_Corporate', 'sector_HigherEd',
       'sector_PreK-12'],
      dtype='object')
```

In [50]:

```
products_info.head()
```

Out[50]:

	LP ID	URL	Product Name	Provider/Company Name	Primary Essential Function	sector_Cor
0	13117	https://www.splashmath.com	SplashLearn	StudyPad Inc.	LC - Digital Learning Platforms	
1	66933	https://abcmouse.com	ABCmouse.com	Age of Learning, Inc	LC - Digital Learning Platforms	
2	50479	https://www.abcya.com	ABCya!	ABCya.com, LLC	LC - Sites, Resources & Reference - Games & Si...	
3	92993	http://www.aleks.com/	ALEKS	McGraw-Hill PreK-12	LC - Digital Learning Platforms	
4	73104	https://www.achieve3000.com/	Achieve3000	Achieve3000	LC - Digital Learning Platforms	

'Primary Essential Function' 컬럼을 기본 및 하위 범주로 분할

In [51]:

```
products_info['pri_function_main'] = products_info['Primary Essential Function'].apply(lambda x: x.
products_info['pri_function_sub'] = products_info['Primary Essential Function'].apply(lambda x: x.s
```

In [52]:

```
products_info.head()
```

Out[52]:

	LP ID	URL	Product Name	Provider/Company Name	Primary Essential Function	sector_Cor
0	13117	https://www.splashmath.com	SplashLearn	StudyPad Inc.	LC - Digital Learning Platforms	
1	66933	https://abcmouse.com	ABCmouse.com	Age of Learning, Inc	LC - Digital Learning Platforms	
2	50479	https://www.abcya.com	ABCya!	ABCya.com, LLC	LC - Sites, Resources & Reference - Games & Si...	
3	92993	http://www.aleks.com/	ALEKS	McGraw-Hill PreK-12	LC - Digital Learning Platforms	
4	73104	https://www.achieve3000.com/	Achieve3000	Achieve3000	LC - Digital Learning Platforms	

In [53]:

```
products_info['pri_function_sub'].unique()
```

Out[53]:

```
array(['Digital Learning Platforms', 'Sites, Resources & Reference',
       'Courseware & Textbooks', 'Study Tools', 'Teacher Resources',
       'Learning Management Systems (LMS)', 'Content Creation & Curation',
       'Online Course Providers & Technical Skills Development',
       'Classroom Engagement & Instruction', 'School Management Software',
       'Other', 'Data, Analytics & Reporting', 'Virtual Classroom', nan,
       'Career Planning & Job Search', 'Human Resources',
       'Large-Scale & Standardized Testing',
       'Sites, Resources & References',
       'Admissions, Enrollment & Rostering',
       'Environmental, Health & Safety (EHS) Compliance'], dtype=object)
```

In [54]:

```
# Synchronize similar values
products_info['pri_function_sub'] = products_info['pri_function_sub'].replace(
          {'Sites, Resources & References' : 'Sites, Resources & Reference'})
products_info.drop("Primary Essential Function", axis=1, inplace=True)
```

In [55]:

```
products_info['pri_function_sub'].unique()
```

Out[55]:

```
array(['Digital Learning Platforms', 'Sites, Resources & Reference',
       'Courseware & Textbooks', 'Study Tools', 'Teacher Resources',
       'Learning Management Systems (LMS)', 'Content Creation & Curation',
       'Online Course Providers & Technical Skills Development',
       'Classroom Engagement & Instruction', 'School Management Software',
       'Other', 'Data, Analytics & Reporting', 'Virtual Classroom', nan,
       'Career Planning & Job Search', 'Human Resources',
       'Large-Scale & Standardized Testing',
       'Admissions, Enrollment & Rostering',
       'Environmental, Health & Safety (EHS) Compliance'], dtype=object)
```

데이터 전처리를 통해 아래 컬럼들을 생성함.

In [58]:

```
products_info[ ['sector_Corporate', 'sector_HigherEd', 'sector_PreK-12',
       'pri_function_main', 'pri_function_sub'] ]
```

Out[58]:

	sector_Corporate	sector_HigherEd	sector_PreK-12	pri_function_main	pri_function_sub
0	0	0	1	LC	Digital Learning Platforms
1	0	0	1	LC	Digital Learning Platforms
2	0	0	1	LC	Sites, Resources & Reference
3	0	1	1	LC	Digital Learning Platforms
4	0	0	1	LC	Digital Learning Platforms
...	...	...	...	...	...
367	1	1	1	SDO	Other
368	1	1	1	LC	Content Creation & Curation
369	0	1	1	LC	Sites, Resources & Reference
370	0	0	0	NaN	NaN
371	0	0	0	NaN	NaN

372 rows × 5 columns

engagement_data의 데이터 파일에 district_id를 추가

In [59]:

```
districts_info.district_id.unique()
```

Out[59]:

```
array([8815, 4921, 5987, 3710, 7177, 9812, 6584, 1044, 7457, 1904, 5527,
       2257, 7614, 4808, 1877, 2779, 8328, 8539, 9043, 1549, 4051, 7305,
       2167, 6577, 4602, 4936, 4520, 7785, 3668, 7970, 5231, 9589, 8433,
       2165, 2074, 1142, 7964, 8784, 7798, 3550, 1444, 2601, 7660, 9899,
       1742, 4629, 4569, 4949, 6250, 8425, 6418, 1558, 3222, 1772, 5604,
       9007, 8884, 1712, 3412, 2940, 5042, 3692, 4683, 2567, 2321, 7767,
       7308, 5006, 9140, 8902, 5890, 4031, 6640, 6194, 3864, 2598, 5600,
       2991, 2106, 6919, 7980, 2060, 7387, 1000, 5150, 2956, 9553, 1536,
       8937, 1791, 4516, 2872, 2439, 8520, 2130, 3772, 4775, 9778, 5524,
       1470, 5802, 1324, 3160, 2393, 9230, 3248, 8556, 5627, 4550, 7752,
       2729, 4348, 3986, 9537, 1052, 6762, 3670, 1204, 2870, 3558, 1450,
       3080, 2517, 1570, 4668, 6055, 2285, 2172, 7741, 6998, 3322, 4083,
       3936, 7675, 4744, 9478, 7541, 1270, 8076, 6345, 4183, 9357, 5510,
       6104, 3228, 5422, 8127, 3640, 8256, 1857, 5479, 3314, 8748, 4373,
       7342, 6046, 7723, 5934, 9927, 2441, 6144, 4314, 9536, 6512, 3732,
       2201, 9303, 3266, 1965, 5882, 1705, 9515, 8103, 4929, 7975, 7164],
      dtype=int64)
```

In [61]:

```
PATH = '../../data/learnplatform-covid19-impact/engagement_data'

temp = []

for district in districts_info.district_id.unique():
    df = pd.read_csv(f'{PATH}/{district}.csv', index_col=None, header=0)
    df['district_id'] = district
    temp.append(df)

len(temp)
```

Out[61]:

```
176
```

In [63]:

```
temp[0:1]
```

Out[63]:

```
[             time  lp_id  pct_access  engagement_index  district_id
 0       2020-01-27  32213      100.00           3000.00         8815
 1       2020-02-25  90153       33.33           2666.67         8815
 2       2020-02-25  99916        0.00               NaN         8815
 3       2020-02-25  28504        0.00               NaN         8815
 4       2020-02-25  95731       33.33            333.33         8815
 ...            ...    ...         ...               ...          ...
 134921  2020-12-31  98468        0.07              1.04         8815
 134922  2020-12-31  99984        0.00               NaN         8815
 134923  2020-12-31  90014        0.00               NaN         8815
 134924  2020-12-31  43876        0.00               NaN         8815
 134925  2020-12-31  57084        0.50             37.95         8815

 [134926 rows x 5 columns]]
```

In [64]:

```
engagement = pd.concat(temp)
engagement = engagement.reset_index(drop=True)
engagement.head()
```

Out[64]:

	time	lp_id	pct_access	engagement_index	district_id
0	2020-01-27	32213.0	100.00	3000.00	8815
1	2020-02-25	90153.0	33.33	2666.67	8815
2	2020-02-25	99916.0	0.00	NaN	8815
3	2020-02-25	28504.0	0.00	NaN	8815
4	2020-02-25	95731.0	33.33	333.33	8815

In [65]:

```
engagement.shape
```

Out[65]:

```
(17435744, 5)
```

```
* 대부분의 학군에는 366일의 고유 일수가 있습니다.
* 그러나 43개의 학군의 경우, 366일 미만의 고유한 데이터를 사용할 수 있음.
* district_id 3670의 경우, 2020-02-15부터 2020-03-02의 데이터만 사용이 가능.
* district_id 2872의 경우, 2020년 1월의 데이터만 사용 가능하고, 2월과 3월의 각각 1일, 총 2일만 데이터 사용가능
```

In [77]:

```
len(engagement.district_id.unique())
```

Out[77]:

```
176
```

In [78]:

```
engagement.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 17435744 entries, 0 to 17435743
Data columns (total 5 columns):
 #   Column            Dtype  
---  ------            -----  
 0   time              object 
 1   lp_id             float64
 2   pct_access        float64
 3   engagement_index  float64
 4   district_id       int64  
dtypes: float64(3), int64(1), object(1)
memory usage: 665.1+ MB
```

In [74]:

```
engagement[engagement['district_id']==3670].time.unique()
```

Out[74]:

```
array(['2020-02-15', '2020-02-16', '2020-02-17', '2020-02-18',
       '2020-02-19', '2020-02-20', '2020-02-21', '2020-02-22',
       '2020-02-23', '2020-02-24', '2020-02-25', '2020-02-26',
       '2020-02-27', '2020-02-28', '2020-03-02'], dtype=object)
```

In [75]:

```
engagement[engagement['district_id']==2872].time.unique()
```

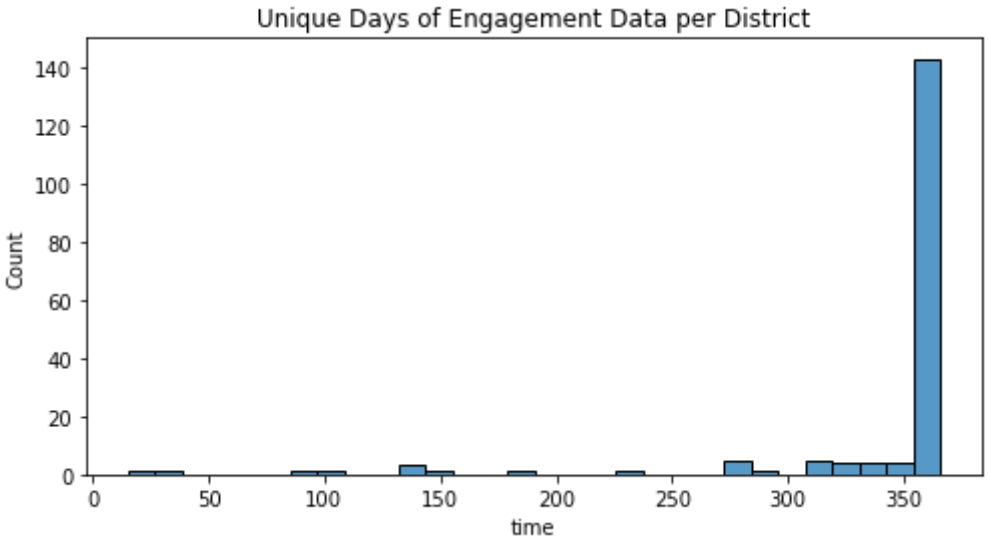
Out[75]:

```
array(['2020-01-01', '2020-01-02', '2020-01-03', '2020-01-04',
       '2020-01-05', '2020-01-06', '2020-01-07', '2020-01-08',
       '2020-01-09', '2020-01-10', '2020-01-11', '2020-01-12',
       '2020-01-13', '2020-01-14', '2020-01-15', '2020-01-16',
       '2020-01-17', '2020-01-18', '2020-01-19', '2020-01-20',
       '2020-01-21', '2020-01-22', '2020-01-23', '2020-01-24',
       '2020-01-25', '2020-01-26', '2020-01-27', '2020-01-28',
       '2020-01-29', '2020-01-30', '2020-01-31', '2020-02-04',
       '2020-03-04'], dtype=object)
```

In [76]:

```
fig, ax = plt.subplots(1, 1, figsize=(8,4))

sns.histplot(engagement.groupby('district_id').time.nunique(), bins=30)
ax.set_title('Unique Days of Engagement Data per District')
plt.show()
```



- 각 district_id에 대해 사용가능한 날짜 카운트를 히스토그램으로 표현

In [ ]:

```
# 앞에서 확인한 engagement를 지우고, 새롭게 만든다.
del engagement

temp = []

for district in districts_info.district_id.unique():
    df = pd.read_csv(f'{PATH}/{district}.csv', index_col=None, header=0)
    df["district_id"] = district
    if df.time.nunique() == 366:
        temp.append(df)

engagement = pd.concat(temp)
engagement = engagement.reset_index(drop=True)
```

In [80]:

```
districts_info.shape, products_info.shape
```

Out[80]:

```
((176, 7), (372, 9))
```

In [81]:

```
# 전체 2020년이 았는 데이터만 합친다.
districts_info = districts_info[districts_info.district_id.isin(engagement.district_id.unique())].re
products_info = products_info[products_info['LP ID'].isin(engagement.lp_id.unique())].reset_index(dr
```

In [82]:

```
districts_info.shape, products_info.shape
```

Out[82]:

```
((176, 7), (369, 9))
```

- 많은 양의 데이터를 제거했다. 이것은 분명 정보의 손실로 이어질 수 있다. 그러나 다른 한편으로 데이터를 쉽게 비교가 가능.
- 2020년에 districts가 없는 지역을 삭제, 데이터가 불완전한 지역을 삭제

In [83]:

```
engagement.time = engagement.time.astype('datetime64[ns]')
```

EDA(Exploratory Data Analysis)

- 이용 가능한 학군 확인.
- 학군이 가장 많이 있는 주는 CT(29)와 UT(24)
- 학군이 하나만 있는 주는 (FL, TN, NY, AZ)

In [84]:

```
us_state_abbrev = {
    'Alabama': 'AL',
    'Alaska': 'AK',
    'American Samoa': 'AS',
    'Arizona': 'AZ',
    'Arkansas': 'AR',
    'California': 'CA',
    'Colorado': 'CO',
    'Connecticut': 'CT',
    'Delaware': 'DE',
    'District Of Columbia': 'DC',
    'Florida': 'FL',
    'Georgia': 'GA',
    'Guam': 'GU',
    'Hawaii': 'HI',
    'Idaho': 'ID',
    'Illinois': 'IL',
    'Indiana': 'IN',
    'Iowa': 'IA',
    'Kansas': 'KS',
    'Kentucky': 'KY',
    'Louisiana': 'LA',
    'Maine': 'ME',
    'Maryland': 'MD',
    'Massachusetts': 'MA',
    'Michigan': 'MI',
    'Minnesota': 'MN',
    'Mississippi': 'MS',
    'Missouri': 'MO',
    'Montana': 'MT',
    'Nebraska': 'NE',
    'Nevada': 'NV',
    'New Hampshire': 'NH',
    'New Jersey': 'NJ',
    'New Mexico': 'NM',
    'New York': 'NY',
    'North Carolina': 'NC',
    'North Dakota': 'ND',
    'Northern Mariana Islands':'MP',
    'Ohio': 'OH',
    'Oklahoma': 'OK',
    'Oregon': 'OR',
    'Pennsylvania': 'PA',
    'Puerto Rico': 'PR',
    'Rhode Island': 'RI',
    'South Carolina': 'SC',
    'South Dakota': 'SD',
    'Tennessee': 'TN',
    'Texas': 'TX',
    'Utah': 'UT',
    'Vermont': 'VT',
    'Virgin Islands': 'VI',
    'Virginia': 'VA',
    'Washington': 'WA',
    'West Virginia': 'WV',
    'Wisconsin': 'WI',
    'Wyoming': 'WY'
}
```

In [85]:

```
districts_info['state_abbrev'] = districts_info['state'].replace(us_state_abbrev)
districts_info_by_state = districts_info['state_abbrev'].value_counts().to_frame().reset_index(drop=
districts_info_by_state.head()
```

Out[85]:

	index	state_abbrev
0	CT	30
1	UT	29
2	MA	21
3	IL	18
4	CA	12

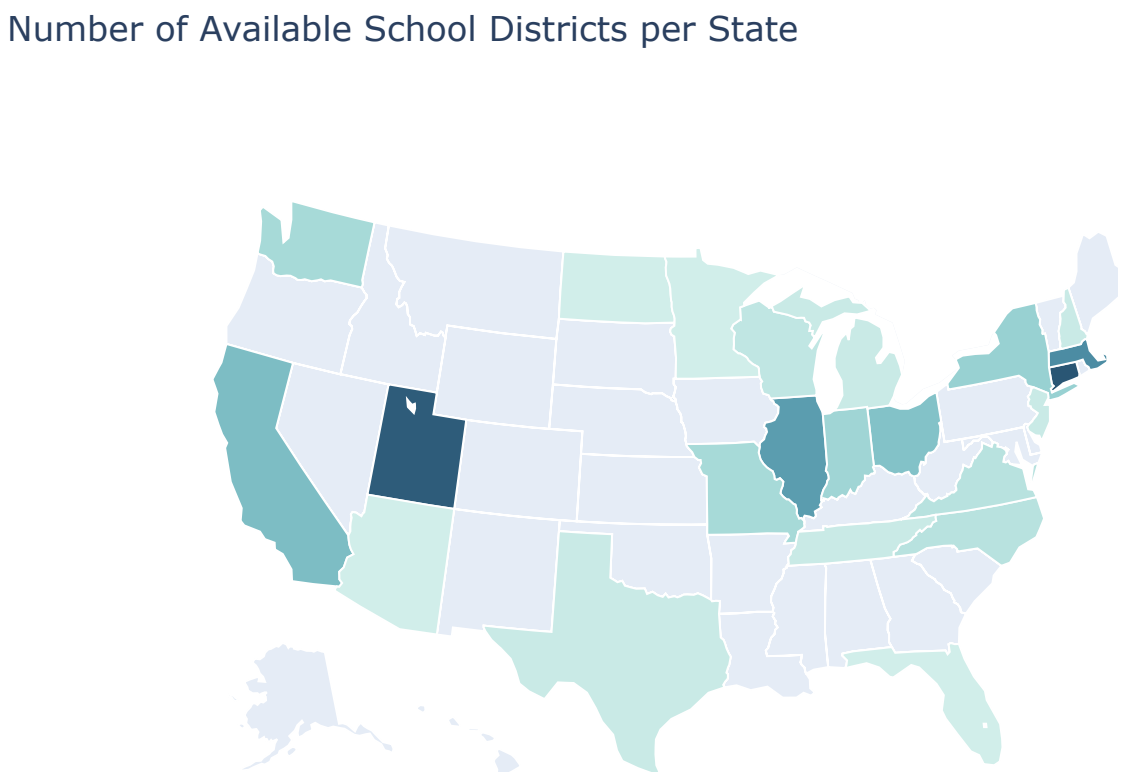
In [86]:

```
districts_info_by_state.columns = ['state_abbrev', 'num_districts']

fig = go.Figure()
layout = dict(
    title_text = "Number of Available School Districts per State",
    geo_scope='usa',
)

fig.add_trace(
    go.Choropleth(
        locations=districts_info_by_state.state_abbrev,
        zmax=1,
        z = districts_info_by_state.num_districts,
        locationmode = 'USA-states', # set of locations match entries in `locations`
        marker_line_color='white',
        geo='geo',
        colorscale=px.colors.sequential.Teal,
    )
)

fig.update_layout(layout)
fig.show()
```



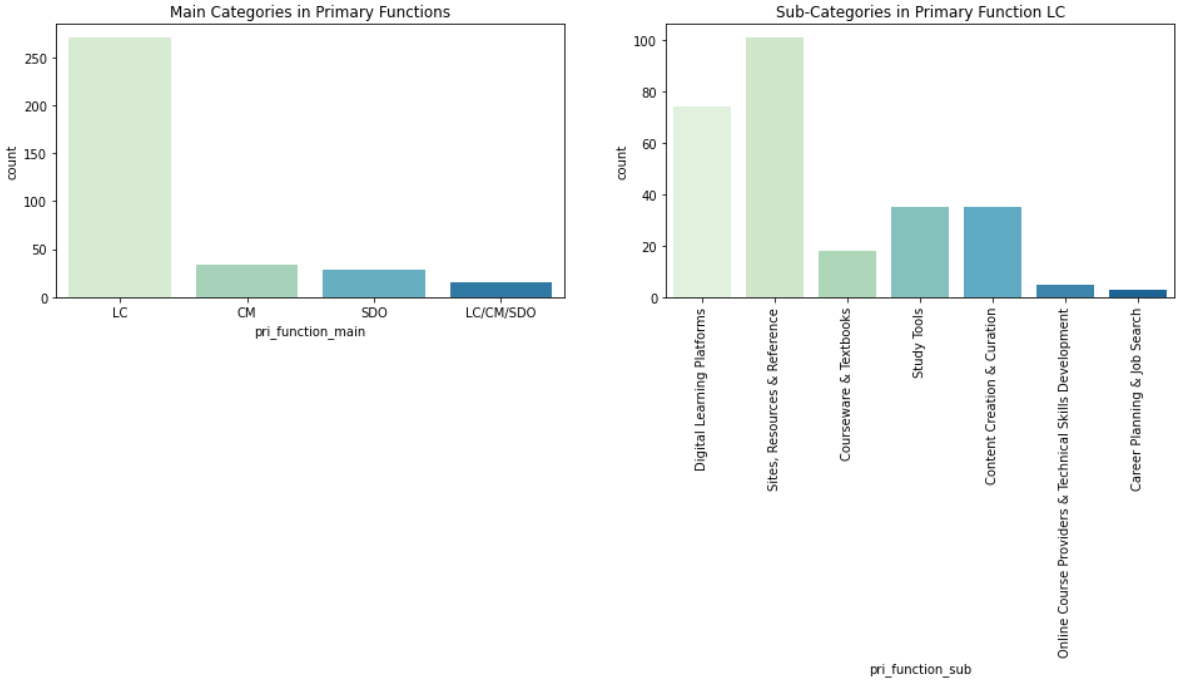
- 'Primary Essential Function'의 열에서 가장 일반적인 범주는 LC(learning & curriculum), 교실 관리(CM)및 학교 및 학군 운영(SDO)

In [88]:

```
fig, ax = plt.subplots(1, 2, figsize=(16,4))
sns.countplot(data=products_info, x='pri_function_main', palette ='GnBu', ax=ax[0])
ax[0].set_title('Main Categories in Primary Functions')

sns.countplot(data=products_info[products_info.pri_function_main == 'LC'],
                      x='pri_function_sub', palette ='GnBu', ax=ax[1])

ax[1].set_title('Sub-Categories in Primary Function LC')
ax[1].set_xticklabels(ax[1].get_xticklabels(), rotation=90)
plt.show()
```



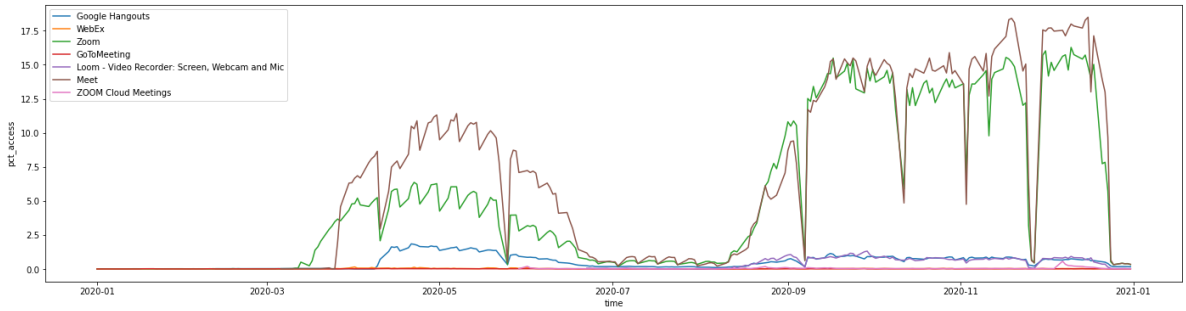
In [89]:

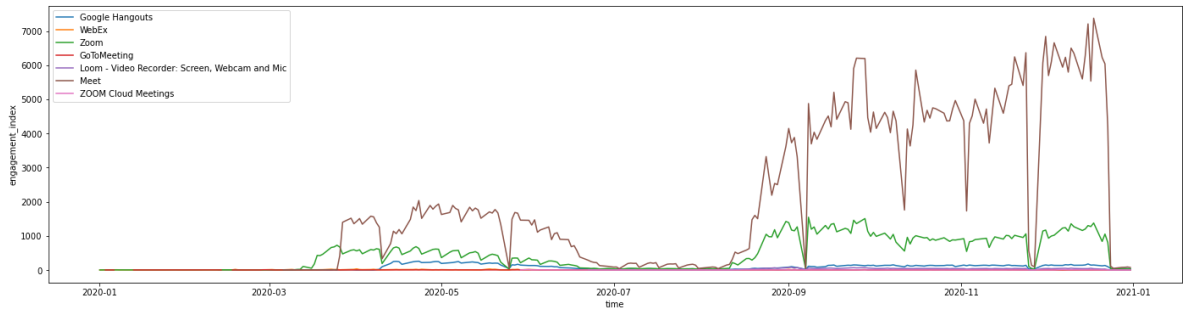
```
virtual_classroom_lp_id = products_info[
                          products_info.pri_function_sub == 'Virtual Classroom']['LP ID'].unique()

# Remove weekends from the dataframe
engagement['weekday'] = pd.DatetimeIndex(engagement['time']).weekday
engagement_without_weekends = engagement[engagement.weekday < 5]

# Figure 1
f, ax = plt.subplots(nrows=1, ncols=1, figsize=(24, 6))
for virtual_classroom_product in virtual_classroom_lp_id:
    temp = engagement_without_weekends[
            engagement_without_weekends.lp_id == virtual_classroom_product
        ].groupby('time').pct_access.mean().to_frame().reset_index(drop=False)
    sns.lineplot(x=temp.time, y=temp.pct_access,
                 label=products_info[
                 products_info['LP ID'] == virtual_classroom_product]['Product Name'].values[0])
plt.legend()
plt.show()

# Figure 2
f, ax = plt.subplots(nrows=1, ncols=1, figsize=(24, 6))
for virtual_classroom_product in virtual_classroom_lp_id:
    temp = engagement_without_weekends[
            engagement_without_weekends.lp_id == virtual_classroom_product
           ].groupby('time').engagement_index.mean().to_frame().reset_index(drop=False)
    sns.lineplot(x=temp.time,
                 y=temp.engagement_index,
                 label=products_info[
                     products_info['LP ID'] == virtual_classroom_product]['Product Name'].values[0])
plt.legend()
plt.show()
```





확인된 내용

- 홈 스쿨링은 3월 초에 시작

- 3월과 7월 사이에 종모양이 있음.
- 7월과 8월에는 여름 방학이 있으므로 참석할 수업이 없음.
- 여름 방학 이후 pct_access는 전염병 초기에 관찰된 대로 더 높은 수준으로 증가 다소 일정하게 유지
- 연중 내내 pct_access에 몇몇 하락이 모인다. 이는 공휴일 또는 기타 공휴일 가능성이 있음.
- Zoom과 Meet는 가상 교실에서 가장 인기 있는 제품.

In [91]:

```
products_info.head()
```

Out[91]:

	LP ID	URL	Product Name	Provider/Company Name	sector_Corporate	sec
0	13117	https://www.splashmath.com	SplashLearn	StudyPad Inc.	0	
1	66933	https://abcmouse.com	ABCmouse.com	Age of Learning, Inc	0	
2	50479	https://www.abcya.com	ABCya!	ABCya.com, LLC	0	
3	92993	http://www.aleks.com/	ALEKS	McGraw-Hill PreK-12	0	
4	73104	https://www.achieve3000.com/	Achieve3000	Achieve3000	0	

In [90]:

```
display(products_info.sum())

display(products_info.groupby('pri_function_main')['pri_function_sub'].value_counts().to_frame())
```

```
LP ID                                                    20136352
URL                  https://www.splashmath.comhttps://abcmouse.com... (https://
www.splashmath.comhttps://abcmouse.com...)
Product Name         SplashLearnABCmouse.comABCya!ALEKSAchieve3000A...
Provider/Company Name    StudyPad Inc.Age of Learning, Inc ABCya.com, L...
sector_Corporate                                                     115
sector_HigherEd                                                      179
sector_PreK-12                                                       348
dtype: object
```

		pri_function_sub
pri_function_main	**pri_function_sub**	
CM	**Classroom Engagement & Instruction**	20
	Teacher Resources	7
	Virtual Classroom	7
LC	**Sites, Resources & Reference**	101
	Digital Learning Platforms	74
	Content Creation & Curation	35
	Study Tools	35
	Courseware & Textbooks	18
	Online Course Providers & Technical Skills Development	5
	Career Planning & Job Search	3
LC/CM/SDO	**Other**	16
SDO	**Data, Analytics & Reporting**	11
	Learning Management Systems (LMS)	5
	Human Resources	4
	School Management Software	4
	Large-Scale & Standardized Testing	2
	Admissions, Enrollment & Rostering	1
	Environmental, Health & Safety (EHS) Compliance	1
	Other	1

Summary

- 달성하고자 하는 바에 따라 신중하게 구역을 사전 선택하는 것이 좋다.
- 노트북에서 접근하는 방식이 개인의 목적에 맞지 않을 수도 있음.
- 디지털 학습을 볼 때 실제로 디지털 학습을 적용한 학군을 파악하는데 시간을 할애할 수 있음.

In [ ]: