

1. choose publicly available data

1.1 공공 데이터 포털([공공데이터포털 \(data.go.kr\)](https://www.data.go.kr/))

- 소방청_연간화재통계(<https://www.data.go.kr/data/15060386/fileData.do>)
- 소방청_화재발생정보(<https://www.data.go.kr/data/15044003/fileData.do>)

2. Explain the details of data

2.1 소방청_연간화재통계(파일명: 소방청_연간화재통계_20211231.csv)

- 2021년 한 해 발생한 전국 화재에 대한 통계 데이터
- 시도, 시군구, 읍면동, 화재 유형, 발화 열원, 발화 요인, 최초 착화물, 인명피해 현황, 재산피해 현황, 장소에 대한 데이터
- 소방청_화재발생정보 데이터와 전반적으로 같으나, 발화 열원, 발화 요인, 최초 착화물과 관련된 정보를 추가로 가지고 있음

2.1.1 데이터 컬럼 소개

컬럼명	컬럼 설명	도메인 분류	데이터 타입
화재발생년월일	화재가 발생한 연도 및 연월일, 시, 분	날짜/시간_연월일시분	object
시도	화재가 발생한 곳의 시, 도	명칭_명	object
시군구	화재가 발생한 곳의 위치(시, 군, 구)	명칭_명	object
화재유형	화재의 유형(건축, 구조물, 임야, 기타 등)	명칭_명	object
발화열원대분류	화재가 일어나게 된 열의 공급원을 크게 나누어 가름	명칭_명	object
발화열원소분류	화재가 일어나게 된 열의 공급원을 상세하게 나누어 가름	명칭_명	object
발화요인대분류	화재를 일으킨 주요 원인을 크게 구분함	내용_내용	object
발화요인소분류	화재를 일으킨 주요 원인을 작게 구분함	내용_내용	object
최초착화물대분류	최초의 두 자리 이상의 리간드가 중심 금속 원자와 배위 결합하여 고리 모양을 이룬 착화합물을 작게 구분	명칭_명	object
최초착화물소분류	최초의 두 자리 이상의 리간드가 중심 금속 원자와 배위 결합하여 고리 모양을 이룬 착화합물을 크게 구분	명칭_명	object
인명피해(명)소계	화재로 인한 인명피해만을 더한 합계	수량_수	int64
사망	화재로 인한 사망자의 수	수량_수	int64
부상	화재로 인한 부상자의 수	수량_수	int64
재산피해소계	화재로 인한 재산피해액만을 더한 합계	수량_수	int64
장소대분류	화재가 일어났던 곳을 크게 구분	명칭_명	object
장소중분류	화재가 일어난 장소를 대분류보다 자세하고 소분류 보다 넓은 범위로 가름	명칭_명	object
장소소분류	화재가 일어났던 곳을 작게 구분	명칭_명	object

* bold체로 표시된 행은 소방청_화재발생정보 데이터와 중복되지 않는 소방청_연간화재통계에만 존재하는 컬럼을 의미함

2.2 소방청_화재발생정보(파일명: 소방청_화재발생 정보_20211231.csv)

- 2021년 한 해 발생한 전국 화재 발생 데이터
- 시도별 화재발생일, 화재발생일, 화재 발생 주소, 화재 원인, 인명피해, 재산 피해와 관련된 데이터
- 소방청_연간화재통계 데이터와 전반적으로 같으나, 재산 피해와 관련된 자세한 정보(부동산, 동산)를 추가로 가지고 있음

2.2.1 데이터 컬럼 소개(공공데이터포털)

컬럼명	컬럼 설명	도메인 분류	데이터 타입
화재발생년월일	화재가 발생한 연도 및 연월일, 시, 분	날짜/시간_연월일시분	object
시도	화재가 발생한 곳의 시, 도	명칭_명	object
시군구	화재가 발생한 곳의 위치(시, 군, 구)	명칭_명	object
인명피해(명)소계	화재로 인한 인명피해만을 더한 합계	수량_수	int64
사망	화재로 인한 사망자의 수	수량_수	int64
부상	화재로 인한 부상자의 수	수량_수	int64
재산피해소계	화재로 인한 재산피해액만을 더한 합계	수량_수	int64
부동산	화재로 인한 부동산 재산 피해액	수량_수	int64
동산	화재로 인한 동산 재산 피해액	수량_수	int64
장소대분류	화재가 일어났던 곳을 크게 구분	명칭_명	object
장소중분류	화재가 일어난 장소를 대분류보다 자세하고 소분류 보다 넓은 범위로 가름	명칭_명	object
장소소분류	화재가 일어났던 곳을 작게 구분	명칭_명	object

* bold체로 표시된 행은 소방청_연간화재통계와 중복되지 않는 소방청_화재발생정보 데이터에만 존재하는 컬럼을 의미함

2.3 데이터 미리보기

2.3.1 소방청_연간화재통계

	화재발생 년월일	시도	시군구	화재유 형	발화열 원대분류	발화열원소 분류	발화요 인대분류	발화요 인소분류	최초착화 물대분류	최초착화 물소분류	인명피해 (명)소계	사 망	부 상	재산피 해소계	장소대 분류	장소 중분류	장소소 분류
0	2021-01-01 00:00	서울특별시	구로구	건축, 구조물	불꽃, 불티	용접, 절단, 연마	부주의	용접, 절단, 연마	기타	기타	0	0	0	50	산업시설	공장시설	금속기계 및 기구공업
1	2021-01-01 00:05	광주광역시	광산구	건축, 구조물	작동기기	기기 전도, 복사열, 기기발열	부주의	음식물 조리중	식품	튀김유	0	0	0	146	산업시설	공장시설	전기, 전자공업
2	2021-01-01 00:06	광주광역시	광산구	건축, 구조물	미상	미상	미상	미상	미상	미상	1	1	0	5148	판매, 업무시설	일반업무	오피스텔

2.3.2 소방청_화재발생 경보

	화재발생년월일	시도	시군구	인명피해소계	사망	부상	재산피해소계	부동산	동산	장소대분류	장소중분류	장소소분류
0	2021-01-01 00:00	서울특별시	구로구	0	0	0	50	0.0	50.0	산업시설	공장시설	금속기계 및 기구공업
1	2021-01-01 00:05	광주광역시	광산구	0	0	0	146	86.0	60.0	산업시설	공장시설	전기, 전자공업
2	2021-01-01 00:06	광주광역시	광산구	1	1	0	5148	5111.0	37.0	판매, 업무시설	일반업무	오피스텔
3	2021-01-01 00:07	경기도	여주시	0	0	0	62588	35620.0	26968.0	생활서비스	음식점	한식
4	2021-01-01 00:12	경기도	양평군	0	0	0	118081	87509.0	30572.0	산업시설	동식물시설	계사

2.4 데이터 결합 및 중복 칼럼 제거

2.4.1 데이터 결합

```
data1=pd.read_csv("소방청_연간화재통계_20211231.csv", encoding="CP949")
data2=pd.read_csv("소방청_화재발생 정보_20211231.csv", encoding="CP949")
df=pd.concat([data1, data2], axis = 1)
df.head(5)
```

	화재발 생년월일	시도	시군구	화재 유형	발화 열원 대분류	발화 열원 소분류	발화 요인 대분류	발화 요인 소분류	최초착 화물대 분류	최초 작화 물소 분류	...	시군구	인명 피해 소계	사 망	부 상	재산피 해소계	부동산	동산	장소 대분류	장소 중분류	장소소 분류
0	2021-01-01 00:00	서울특별시	구로구	건축, 구조물	불꽃, 불티	응접, 절단, 연마	부주의	응접, 절단, 연마	기타	기타	...	구로구	0	0	0	50	0.0	50.0	산업 시설	공장 시설	금속기계 및 기구공업
1	2021-01-01 00:05	광주광역시	광산구	건축, 구조물	작동 기기	기기 전도, 복사열, 기기 발열	부주의	음식물 조리중	식품	튀김 유	...	광산구	0	0	0	146	86.0	60.0	산업 시설	공장 시설	전기, 전자공업
2	2021-01-01 00:06	광주광역시	광산구	건축, 구조물	미상	미상	미상	미상	미상	미상	...	광산구	1	1	0	5148	5111.0	37.0	판매, 업무 시설	일반무	오피스텔
3	2021-01-01 00:07	경기도	여주시	건축, 구조물	작동 기기	전기적 아크(단락)	전기적 요인	접촉불량에 의한 단락	전기, 전자	전선 피복	...	여주시	0	0	0	62588	35620.0	26968.0	생활 서비스	음식점	한식
4	2021-01-01 00:12	경기도	양평군	건축, 구조물	작동 기기	전기적 아크(단락)	전기적 요인	미확인 단락	전기, 전자	전선 피복	...	양평군	0	0	0	118081	87509.0	30572.0	산업 시설	동식물 시설	계사

data1(소방청_연간화재통계)와 data2(소방청_화재발생 정보)를 컬럼을 기준으로 데이터를 결합하였다. data1과 data2에는 중복되는 컬럼이 존재하므로, 생성된 df에 중복 컬럼을 제거하는 것이 필요하다.

코드)

```
data1=pd.read_csv("소방청_연간화재통계_20211231.csv", encoding="CP949")
data2=pd.read_csv("소방청_화재발생 정보_20211231.csv", encoding="CP949")
df=pd.concat([data1, data2], axis = 1)
df.head(5)
```

2.4.2 중복 컬럼 제거

```
df.T.duplicated()
```

```
화재발생년월일    False
시도              False
시군구            False
화재유형          False
발화열원대분류    False
발화열원소분류    False
발화요인대분류    False
발화요인소분류    False
최초착화물대분류  False
최초착화물소분류  False
인명피해(명)소계  False
사망              False
부상              False
재산피해소계      False
장소대분류        False
장소중분류        False
장소소분류        False
화재발생년월일    True
시도              True
시군구            True
인명피해소계      True
사망              True
부상              True
재산피해소계      True
부동산            False
동산              False
장소대분류        True
장소중분류        True
장소소분류        True
dtype: bool
```

```
df=df.loc[:,~df.T.duplicated()]
```

df을 행과 열을 전치(transpose)하고, duplicated()를 통해 중복되는 컬럼을 확인한 뒤, 중복된 컬럼을 df.loc로 조건을 부여하여(~df.T.duplicated()) 제거하였다.

코드)

```
df.T.duplicated()

df=df.loc[:,~df.T.duplicated()]
df.info()
```

3. Data preprocessing & Data manipulation

3.1 컬럼명 변경(Renaming column names) - [Data preprocessing1](#)

1) 컬럼명 확인

```
df.columns

Index(['화재발생년월일', '시도', '시군구', '화재유형', '발화열원대분류', '발화열원소분류', '발화요인대분류',
       '발화요인소분류', '최초착화물대분류', '최초착화물소분류', '인명피해(명)소계', '사망', '부상', '재산피해소계',
       '장소대분류', '장소중분류', '장소소분류', '부동산', '동산'],
      dtype='object')
```

2) 컬럼명 변경

```
df.rename(columns={'화재발생년월일': '화재시각',
                  '발화열원대분류': '발화열원(대)',
                  '발화열원소분류': '발화열원(소)',
                  '발화요인대분류': '발화요인(대)',
                  '발화요인소분류': '발화요인(소)',
                  '최초착화물대분류': '최초착화물(대)',
                  '최초착화물소분류': '최초착화물(소)',
                  '인명피해(명)소계': '인명피해',
                  '재산피해소계': '재산피해',
                  '장소대분류': '장소(대)',
                  '장소중분류': '장소(중)',
                  '장소소분류': '장소(소)'}, inplace=True)

df.columns

Index(['화재시각', '시도', '시군구', '화재유형', '발화열원(대)', '발화열원(소)', '발화요인(대)', '발화요인(소)',
       '최초착화물(대)', '최초착화물(소)', '인명피해', '사망', '부상', '재산피해', '장소(대)', '장소(중)',
       '장소(소)', '부동산', '동산'],
      dtype='object')
```

데이터의 컬럼명은 Data preprocessing이나 Data manipulate를 위해, 적절한 길이나 형태로 변경하는 것이 편리할 수 있다.

코드)

```
df.columns

df.rename(columns={'화재발생년월일': '화재시각',
                  '발화열원대분류': '발화열원(대)',
                  '발화열원소분류': '발화열원(소)',
                  '발화요인대분류': '발화요인(대)',
                  '발화요인소분류': '발화요인(소)',
                  '최초착화물대분류': '최초착화물(대)',
                  '최초착화물소분류': '최초착화물(소)',
                  '인명피해(명)소계': '인명피해',
                  '재산피해소계': '재산피해',
                  '장소대분류': '장소(대)',
                  '장소중분류': '장소(중)',
                  '장소소분류': '장소(소)'}, inplace=True)

df.columns
```

3.2 새로운 컬럼 생성(Creating Derive variable(=column)) - Data preprocessing2

1) '화재시각' 컬럼을 이용해, 'month', 'day', 'hour' 컬럼 생성 - Data manipulation1

```
df['화재시각'] = pd.to_datetime(df['화재시각'])
for df in [df]:
    df['month'] = df['화재시각'].dt.month
    df['day'] = df['화재시각'].dt.day
    df['hour'] = df['화재시각'].dt.hour
df.loc[:,['화재시각', 'month', 'day', 'hour']].head()
```

	화재시각	month	day	hour
0	2021-01-01 00:00:00	1	1	0
1	2021-01-01 00:05:00	1	1	0
2	2021-01-01 00:06:00	1	1	0
3	2021-01-01 00:07:00	1	1	0
4	2021-01-01 00:12:00	1	1	0

Data visualization에 이용하기 위해, 날짜형식의 '화재시각' 컬럼을 이용하여 월, 일, 시를 의미하는 'month', 'day', 'hour' 컬럼을 각각 생성하였다.

코드)

```
df['화재시각'] = pd.to_datetime(df['화재시각'])
for df in [df]:
    df['month'] = df['화재시각'].dt.month
    df['day'] = df['화재시각'].dt.day
    df['hour'] = df['화재시각'].dt.hour
df.loc[:,['화재시각', 'month', 'day', 'hour']].head()
```

3.3 불필요한 컬럼 제거(Dropping unnecessary columns) - Data preprocessing3

1) '화재시각', '장소(중)', '장소(소)' 컬럼 제거

```
df=df.drop(columns=['화재시각', '장소(중)', '장소(소)'])
df.columns

Index(['시도', '시군구', '화재유형', '발화열원(대)', '발화열원(소)', '발화요인(대)', '발화요인(소)',
       '최초착화물(대)', '최초착화물(소)', '인명피해', '사망', '부상', '재산피해', '장소(대)', '부동산',
       '동산', 'month', 'day', 'hour'],
      dtype='object')
```

새로운 컬럼인 'month', 'day', 'hour'를 생성하는데 이용한 '화재시각' 컬럼을 삭제하고, Data visualization에 사용하지 않는 '최초착화물(대)', '최초착화물(소)' 컬럼도 추가로 삭제하였다.

코드)

```
df=df.drop(columns=['화재시각', '최초착화물(대)', '최초착화물(소)'])
df.columns
```

3.4 이상치 제거(Outlier detection) - Data preprocessing4

1) '재산 피해' 컬럼의 이상치 확인

```
fig = px.box(df, y='재산피해', notched=True)
fig.update_layout(
    title_text="재산피해의 boxplot",
    title_font=dict(size=13),
    title_x=0.5
)
fig.show()
```



2) 이상치 제거 방안 - Data manipulation2

```
df.sort_values('재산피해', ascending=False)['재산피해'].head(15)
```

```
18196    474323261
11617     37747126
15838     14571586
31182      8831129
19490      7824397
10970      6548228
20745      5912804
24314      5820689
2487       4952216
9560       4936054
9068       4862749
28232      4808902
9550       4680999
31868      4629236
17521      4589061
Name: 재산피해, dtype: int64
```

'재산피해' 컬럼에 대한 boxplot을 통해, 이상치로 추정되는 한 개의 point를 확인할 수 있었다. 이상치 제거를 위해, '재산피해' 컬럼을 기준으로 내림차순으로 정렬하여 값들을 확인하였고, 가장 큰 값인 4742361을 포함한 8개의 값을 이상치로 결정할 수 있었다.

3) 이상치 제거 후, '재산피해' 컬럼의 boxplot 확인

```
df.drop(df[df['재산피해'] >= 5820689].index, inplace = True)
fig = px.box(df, y='재산피해', notched=True)
fig.update_layout(
    title_text="재산피해의 boxplot",
    title_font=dict(size=13),
    title_x=0.5
)
fig.show()
```



이상치로 판단되는 값들 중 가장 작은 값인, 5820689을 기준으로 큰 값들의 인덱스를 drop문을 통해 제거하였다.

이상치 제거 후에 boxplot을 확인하였을 때, 이상치 기준인 제3분위 수+1.5*IQR을 넘는 값들이 존재하는 것으로 보인다. 하지만 이상치로 생각될 수 있는 값들이 동떨어져 있지 않고, 연속적으로 나타나 있어 유의미하다고 생각하여 제거하지 않았다.

코드)

```
fig = px.box(df, y='재산피해', notched=True)
fig.update_layout(
    title_text="재산피해의 boxplot",
    title_font=dict(size=13),
    title_x=0.5
)
fig.show()
```

```
df.sort_values('재산피해', ascending=False)['재산피해'].head(15)
```

```
df.drop(df[df['재산피해'] >= 5820689].index, inplace = True)
fig = px.box(df, y='재산피해', notched=True)
fig.update_layout(
    title_text="재산피해의 boxplot",
    title_font=dict(size=13),
    title_x=0.5
)
fig.show()
```


3.5 결측치 대체(Data imputation) - Data preprocessing5

1) 결측치 확인

```
df.isna().sum()
```

```
화재발생년월일    0
시도              0
화재유형          0
발화열원대분류    0
발화요인대분류    0
최초착화물대분류  0
인명피해(명)소계  0
사망              0
부상              0
재산피해소계      0
장소대분류        0
부동산           2829
동산             2829
dtype: int64
```

2) 결측치 대체 방안 고려 - Data manipulation3

```
df1=df.loc[(df['부동산'].isna() != True)&(df['동산'].isna() != True), ['재산피해', '부동산', '동산']]
df1['부동산+동산']=df1['부동산']+df1['동산']
df1.head()
```

	재산피해	부동산	동산	부동산+동산
0	50	0.0	50.0	50.0
1	146	86.0	60.0	146.0
2	5148	5111.0	37.0	5148.0
3	62588	35620.0	26968.0	62588.0
4	118081	87509.0	30572.0	118081.0

```
df1.loc[df1['재산피해']!=df1['부동산+동산']]
```

재산피해	부동산	동산	부동산+동산
------	-----	----	--------

```
property=df.loc[(df['부동산'].isna() == True)&(df['동산'].isna() == True), ['재산피해', '부동산', '동산']]
property.describe()
```

	재산피해	부동산	동산
count	2829.0	0.0	0.0
mean	0.0	NaN	NaN
std	0.0	NaN	NaN
min	0.0	NaN	NaN
25%	0.0	NaN	NaN
50%	0.0	NaN	NaN
75%	0.0	NaN	NaN
max	0.0	NaN	NaN

‘부동산’, ‘동산’ 컬럼에 2829개의 결측치가 존재함을 확인할 수 있었다.

2)의 과정을 통해, ‘재산피해’ 컬럼이 ‘부동산’, ‘동산’ 컬럼의 합으로 이루어졌음을 알 수 있었다.

이를 통해, ‘부동산’이나 ‘동산’ 컬럼의 결측값이 같은 행에 존재하지 않는다면 ‘재산피해’ 컬럼을 통해 결측치를 채우고, 같은 행에 존재한다면 데이터를 제거하도록 방안을 정하였다.

loc함수와 describe를 통해, ‘부동산’과 ‘동산’ 컬럼은 모든 결측치가 같은 행(2829개 행)에 존재함이 확인되었다.

미리 정해둔 방안에 따르면, 데이터를 제거하는 것이 적절하나 같은 행에 ‘재산피해’ 컬럼 값이 모두 0을 가지므로 ‘부동산’, ‘동산’의 값은 0을 갖는 것이 더 바람직하다고 판단된다.

3) 결측치 대체

```
df['부동산']=df['부동산'].fillna(0).astype(float)
df['동산']=df['동산'].fillna(0).astype(float)
```

따라서 ‘부동산’과 ‘동산’의 결측치는 0으로 대체하였다.

코드)

```
df.isna().sum()
```

```
df1=df.loc[(df['부동산'].isna() != True)&(df['동산'].isna() != True), ['재산피해', '부동산', '동산']]
```

```
df1['부동산+동산']=df1['부동산']+df1['동산']
```

```
df1.head()
```

```
df1.loc[df1['재산피해']!=df1['부동산+동산']]
```

```
property=df.loc[(df['부동산'].isna() == True)&(df['동산'].isna() == True), ['재산피해', '부동산', '동산']]
```

```
property.describe()
```

```
df['부동산']=df['부동산'].fillna(0).astype(float)
```

```
df['동산']=df['동산'].fillna(0).astype(float)
```

```
df.isna().sum()
```

4. Data visualization & Data manipulation

4.0 데이터 시각화의 목적(purpose of Data visualization)

- 화재 유형, 화재 장소, 화재 발생 지역에 따른 재산피해, 발화열원, 사망자 수 등을 시각화하여, 비교하고 화재의 특징을 찾아내고자 함
- 데이터셋에는 범주형 데이터로 이루어진 컬럼(변수)이 대다수이므로, 시각화할 때 보기에 용이하도록 컬럼값의 unique 수를 고려해야 함

1) 범주형 변수의 unique한 값의 갯수

```
for i in df.select_dtypes(include='object').columns:  
    print(i, " : ", df[i].nunique())
```

```
시도 : 17  
시군구 : 228  
화재유형 : 6  
발화열원(대) : 9  
발화열원(소) : 28  
발화요인(대) : 12  
발화요인(소) : 49  
장소(대) : 15  
장소(중) : 50  
장소(소) : 315
```

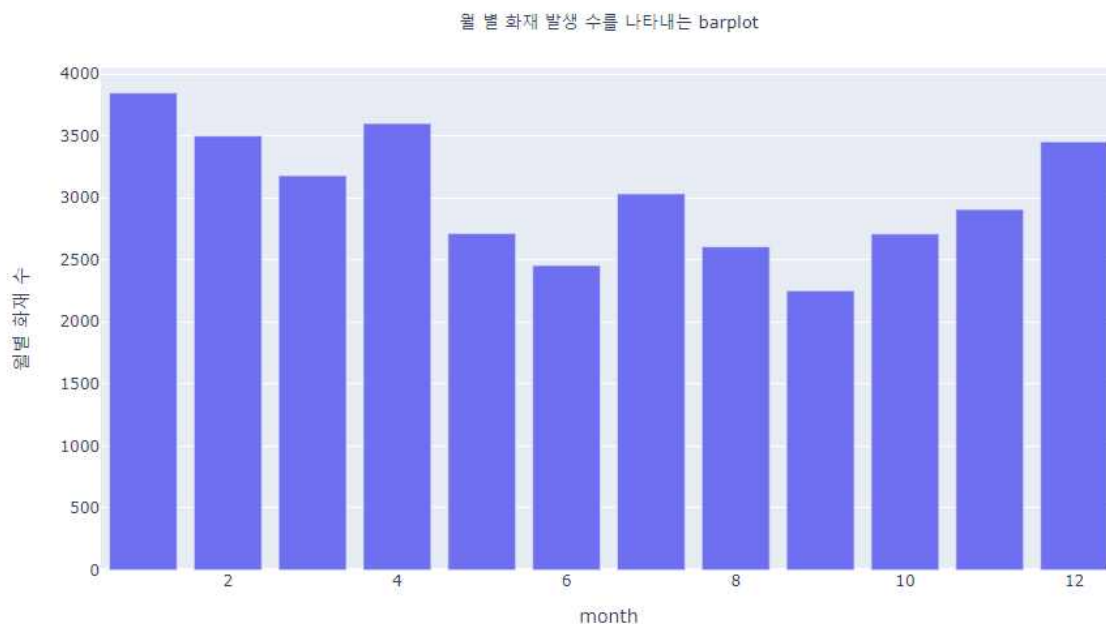
코드)

```
for i in df.select_dtypes(include='object').columns:  
    print(i, " : ", df[i].nunique())
```

4.1 Barplot

1) 월별 화재 발생 수를 나타내는 barplot

```
fire_count = df['month'].value_counts().reset_index()  
fire_count.columns = ['month', '월별 화재 수']  
  
fig = px.bar(fire_count, x='month', y='월별 화재 수')  
fig.update_layout(  
    title_text="월 별 화재 발생 수를 나타내는 barplot",  
    title_font=dict(size=13),  
    title_x=0.5  
)  
fig.show()
```



위의 Barplot은 'month' 컬럼을 이용해 월별 화재 발생 건수를 나타내고 있다.

1월에 3847건으로 가장 많은 화재가 발생하였고, 9월에 2252건으로 가장 적게 화재가 발생하였다.

또한 plot을 통해, 12~4월 사이에 많은 화재가 발생하고, 6~9월에 화재가 적게 발생하는 것을 확인할 수 있다. 이는 습도가 하나의 요인으로 작용할 수 있다고 생각되며, 건조한 겨울, 봄(12~4월)에 화재 가능성이 크고, 습한 여름(6~9월)에 화재 가능성이 작다고 판단될 수 있을 것이다.

코드)

```
fire_count = df['month'].value_counts().reset_index()
fire_count.columns = ['month', '월별 화재 수']
```

```
fig = px.bar(fire_count, x='month', y='월별 화재 수')
fig.update_layout(
    title_text="월 별 화재 발생 수를 나타내는 barplot",
    title_font=dict(size=13),
    title_x=0.5
)
fig.show()
```

4.2 Scatterplot

1) 화재유형에 따른 재산피해 분포를 나타내는 scatter plot

```
fig = px.scatter(df, x="화재유형", y="재산피해", hover_name="장소(소)")
fig.update_layout(
    title_text="화재유형에 따른 재산피해 분포를 나타내는 scatter plot",
    title_font=dict(size=13),
    title_x=0.5
)
fig.show()
```



위의 scatter plot은 화재유형에 따른 재산피해의 분포를 나타낸다.

많은 화재가 건축·구조물의 화재 유형으로 발생하는 것으로 보이고, 그에 따라 재산 피해(액)도 다양한 범위를 갖는 것으로 확인된다. 재산피해가 약 4.95M으로 가장 큰 화재 또한 건축·구조물의 화재 유형인 것을 알 수 있다.

이외의 화재유형에선 화재로 인한 재산피해가 대부분 1M이하로 크게 차이가 없는 것으로 보인다.

그러나 선박·항공기의 화재 유형에서 재산피해가 약 4.93M으로 모든 화재유형을 포함한 재산피해 규모 중 2번째로 큰 화재가 존재한다는 것을 유의할 필요가 있어 보인다.

결과적으로 재산피해 규모가 큰 화재가 건축·구조물의 화재 유형에서 대부분 발생하고 이에 따라 재산피해 규모액의 높은 비율을 차지한다는 것을 간접적으로 알 수 있었다.

코드)

```
fig = px.scatter(df, x="화재유형", y='재산피해', hover_name="장소(소)")
fig.update_layout(
    title_text="화재유형에 따른 재산피해 분포를 나타내는 scatter plot",
    title_font=dict(size=13),
    title_x=0.5
)
fig.show()
```

4.3 Violinplot

1) 화재유형별 인명피해 분포에 대한 violin plot

```
fig = px.violin(df, y="인명피해", x="화재유형", color='화재유형', box=True, points="all", hover_data=df.columns)
fig.update_layout(
    title_text="화재유형 별 인명피해 분포에 대한 violin plot",
    title_font=dict(size=13),
    title_x=0.5
)
fig.show()
```



위의 Violinplot은 화재유형별 인명피해의 분포를 나타낸다.

6개의 화재유형에서 건축,·구조물에 대한 인명피해 분포는 다른 유형들과 확연한 차이를 보인다.

33명으로 가장 많은 인명피해를 발생시킨 화재뿐만 아니라, 다른 화재유형에서 발견할 수 없는 5명 이상의 화재 또한 다수 분포되어 있음을 확인할 수 있다.

추가로 hover_data 옵션을 통해 5명 이상의 인명피해를 발생시킨 화재의 points를 확인한 결과, 주거와 관련된 단독주택과 공동주택의 사례가 많음을 알 수 있었다.

다음으로 기타, 자동차·철도차량, 임야, 선박·항공기, 위험물·가스제조소 등의 유형에서는 기타, 자동차·철도차량, 임야의 화재 유형에서 더 화재 발생이 잦다는 점 외에는 인명피해 분포에 대한 큰 차이점을 발견할 수 없었다.

결론적으로 대부분 화재는 건축,·구조물 유형에서 발생하며, 그중에서도 주거와 관련된 장소에서 인명피해가 큰 화재가 발생한다는 사실을 알 수 있었다.

코드)

```
fig = px.violin(df, y="인명피해", x="화재유형", color='화재유형', box=True, points="all",
    hover_data=df.columns)
fig.update_layout(
    title_text="화재유형 별 인명피해 분포에 대한 violin plot",
    title_font=dict(size=13),
    title_x=0.5
)
fig.show()
```

4.4 Icicle charts

1) 장소별 화재 발생 수를 나타내는 icicle charts

```
fig = px.icicle(df, path=['장소(대)', '장소(중)', '장소(소)'])
fig.update_layout(
    title_text="장소별 화재 발생 수를 나타내는 icicle charts",
    title_font=dict(size=13),
    title_x=0.5
)
fig.show()
```

장소별 화재 발생 수를 나타내는 icicle charts



위의 Icicle charts는 장소를 계층 구조로 나타내고, 장소에 따른 화재 수를 각 블록의 크기로 표현한 것이다. 여기서 ‘장소(대)’ 컬럼은 가장 큰 상위항목이고, 그 뒤로 ‘장소(중)’, ‘장소(소)’ 순서로 하위 항목에 속한다.

‘장소(대)’ 컬럼을 기준으로 확인했을 때, 주거, 기타, 산업시설, 자동차·철도차량 등의 장소 순서대로 화재 발생이 많은 것으로 보인다.

먼저, 주거는 단독주택과 공동주택에서의 화재가 대부분을 차지했고, 공동주택에서는 특히나 아파트에서 화재가 자주 발생하는 것을 볼 수 있었다.

다음으로 기타 장소에서는 야외로 분류되는 장소에서의 화재가 대부분을 차지했고, 야외에서도 특히 쓰레기가 있는 장소로 생각되는 쓰레기와 기타 야외에서 많은 화재가 발생함을 알 수 있었다.

다음으로 산업시설은 공장시설로 분류되는 장소에서의 화재 비중이 가장 컸고, 하위 항목으로 갈수록 다양한 장소에서 비슷한 비율로 화재가 발생했음을 볼 수 있다.

마지막으로 자동차·철도차량에서의 화재는 자동차에서의 화재가 대부분의 비중을 차지하고 있었다. 자동차 중에서도 승용자동차에서 가장 많은 화재가 발생했으며, 다음으로 화물자동차, 오토바이 등의 순서로 많았음을 확인할 수 있었다.

그 뒤로, 생활서비스, 판매·업무시설, 기타서비스 등의 순서로 화재 발생이 많음을 확인할 수 있었다.

코드)

```
fig = px.icicle(df, path=['장소(대)', '장소(중)', '장소(소)'])
fig.update_layout(
    title_text="장소별 화재 발생 수를 나타내는 icicle charts",
    title_font=dict(size=13),
    title_x=0.5
)
fig.show()
```

4.5 Heatmap

1) 주요 장소별 발생한 화재의 발화열원 빈도 수를 저장하는 데이터프레임 생성 - Data mainpulate4

```
living_count=df[df['장소(대)']=='주거']['발화열원(대)'].value_counts().reset_index().sort_values('index')
outside_count=df[df['장소(중)']=='야외']['발화열원(대)'].value_counts().reset_index().sort_values('index')
industry_count=df[df['장소(대)']=='산업시설']['발화열원(대)'].value_counts().reset_index().sort_values('index')
vehicle_count=df[df['장소(중)']=='자동차']['발화열원(대)'].value_counts().reset_index().sort_values('index')
```

4.4)의 결과로 주거, 기타, 산업시설, 자동차·철도차량에서 많은 화재가 발생하는 것을 알 수 있었다.

이에 따라 많은 화재가 발생하는 장소에 대한 주된 발화열원을 알아보고자, heatmap에 사용할 장소별 발화열원의 빈도수를 따로 데이터프레임의 형태로 저장하였다.

* 기타와 자동차·철도차량의 장소는 각각 야외, 자동차가 대부분의 화재 비중을 차지하므로, 기타는 야외로, 자동차·철도차량은 자동차로 장소를 대체함

코드)

```
living_count = df[df['장소(대)']=='주거']['발화열원(대)'].value_counts().reset_index().sort_values('index')
outside_count = df[df['장소(중)']=='야외']['발화열원(대)'].value_counts().reset_index().sort_values('index')
industry_count = df[df['장소(대)']=='산업시설']['발화열원(대)'].value_counts().reset_index().sort_values('index')
vehicle_count = df[df['장소(중)']=='자동차']['발화열원(대)'].value_counts().reset_index().sort_values('index')
```

2) 주요 장소별 발생한 화재의 발화열원 빈도 수 확인

2-1) 주거 장소	2-2) 야외 장소	2-3) 산업 시설	2-4) 자동차																																																																																																																		
<div>living_count</div> <table><tr><th colspan="3">index 발화열원(대)</th></tr><tr><td>5</td><td>기타</td><td>160</td></tr><tr><td>1</td><td>담뱃불,ライター불</td><td>1602</td></tr><tr><td>4</td><td>마찰, 전도, 복사</td><td>823</td></tr><tr><td>3</td><td>미상</td><td>940</td></tr><tr><td>2</td><td>불꽃, 불티</td><td>1211</td></tr><tr><td>7</td><td>자연적 발화열</td><td>27</td></tr><tr><td>0</td><td>작동기기</td><td>5191</td></tr><tr><td>6</td><td>화학적 발화열</td><td>50</td></tr></table>	index 발화열원(대)			5	기타	160	1	담뱃불,ライター불	1602	4	마찰, 전도, 복사	823	3	미상	940	2	불꽃, 불티	1211	7	자연적 발화열	27	0	작동기기	5191	6	화학적 발화열	50	<div>outside_count</div> <table><tr><th colspan="3">index 발화열원(대)</th></tr><tr><td>6</td><td>기타</td><td>102</td></tr><tr><td>0</td><td>담뱃불,ライター불</td><td>2603</td></tr><tr><td>4</td><td>마찰, 전도, 복사</td><td>144</td></tr><tr><td>3</td><td>미상</td><td>318</td></tr><tr><td>1</td><td>불꽃, 불티</td><td>1756</td></tr><tr><td>7</td><td>자연적 발화열</td><td>48</td></tr><tr><td>2</td><td>작동기기</td><td>1142</td></tr><tr><td>8</td><td>폭발물, 폭죽</td><td>16</td></tr><tr><td>5</td><td>화학적 발화열</td><td>143</td></tr></table>	index 발화열원(대)			6	기타	102	0	담뱃불,ライター불	2603	4	마찰, 전도, 복사	144	3	미상	318	1	불꽃, 불티	1756	7	자연적 발화열	48	2	작동기기	1142	8	폭발물, 폭죽	16	5	화학적 발화열	143	<div>industry_count</div> <table><tr><th colspan="3">index 발화열원(대)</th></tr><tr><td>6</td><td>기타</td><td>71</td></tr><tr><td>3</td><td>담뱃불,ライター불</td><td>362</td></tr><tr><td>4</td><td>마찰, 전도, 복사</td><td>296</td></tr><tr><td>2</td><td>미상</td><td>687</td></tr><tr><td>1</td><td>불꽃, 불티</td><td>848</td></tr><tr><td>7</td><td>자연적 발화열</td><td>27</td></tr><tr><td>0</td><td>작동기기</td><td>2623</td></tr><tr><td>8</td><td>폭발물, 폭죽</td><td>2</td></tr><tr><td>5</td><td>화학적 발화열</td><td>218</td></tr></table>	index 발화열원(대)			6	기타	71	3	담뱃불,ライター불	362	4	마찰, 전도, 복사	296	2	미상	687	1	불꽃, 불티	848	7	자연적 발화열	27	0	작동기기	2623	8	폭발물, 폭죽	2	5	화학적 발화열	218	<div>vehicle_count</div> <table><tr><th colspan="3">index 발화열원(대)</th></tr><tr><td>5</td><td>기타</td><td>82</td></tr><tr><td>3</td><td>담뱃불,ライター불</td><td>466</td></tr><tr><td>1</td><td>마찰, 전도, 복사</td><td>678</td></tr><tr><td>2</td><td>미상</td><td>505</td></tr><tr><td>4</td><td>불꽃, 불티</td><td>157</td></tr><tr><td>7</td><td>자연적 발화열</td><td>5</td></tr><tr><td>0</td><td>작동기기</td><td>2127</td></tr><tr><td>6</td><td>화학적 발화열</td><td>24</td></tr></table>	index 발화열원(대)			5	기타	82	3	담뱃불,ライター불	466	1	마찰, 전도, 복사	678	2	미상	505	4	불꽃, 불티	157	7	자연적 발화열	5	0	작동기기	2127	6	화학적 발화열	24
index 발화열원(대)																																																																																																																					
5	기타	160																																																																																																																			
1	담뱃불,ライター불	1602																																																																																																																			
4	마찰, 전도, 복사	823																																																																																																																			
3	미상	940																																																																																																																			
2	불꽃, 불티	1211																																																																																																																			
7	자연적 발화열	27																																																																																																																			
0	작동기기	5191																																																																																																																			
6	화학적 발화열	50																																																																																																																			
index 발화열원(대)																																																																																																																					
6	기타	102																																																																																																																			
0	담뱃불,ライター불	2603																																																																																																																			
4	마찰, 전도, 복사	144																																																																																																																			
3	미상	318																																																																																																																			
1	불꽃, 불티	1756																																																																																																																			
7	자연적 발화열	48																																																																																																																			
2	작동기기	1142																																																																																																																			
8	폭발물, 폭죽	16																																																																																																																			
5	화학적 발화열	143																																																																																																																			
index 발화열원(대)																																																																																																																					
6	기타	71																																																																																																																			
3	담뱃불,ライター불	362																																																																																																																			
4	마찰, 전도, 복사	296																																																																																																																			
2	미상	687																																																																																																																			
1	불꽃, 불티	848																																																																																																																			
7	자연적 발화열	27																																																																																																																			
0	작동기기	2623																																																																																																																			
8	폭발물, 폭죽	2																																																																																																																			
5	화학적 발화열	218																																																																																																																			
index 발화열원(대)																																																																																																																					
5	기타	82																																																																																																																			
3	담뱃불,ライター불	466																																																																																																																			
1	마찰, 전도, 복사	678																																																																																																																			
2	미상	505																																																																																																																			
4	불꽃, 불티	157																																																																																																																			
7	자연적 발화열	5																																																																																																																			
0	작동기기	2127																																																																																																																			
6	화학적 발화열	24																																																																																																																			

2-5) 장소별 발화열원 빈도 수를 배열 형태로 생성

```
data=[[ 160, 1602, 823, 940, 1211, 27, 5191, 0, 50],
       [ 102, 2603, 144, 318, 1756, 48, 1142, 16, 143],
       [ 71, 362, 296, 687, 848, 27, 2623, 2, 218],
       [ 82, 466, 678, 505, 157, 5, 2127, 0, 24]]
```

주요 장소별 발화열원의 빈도수를 데이터프레임으로 직접 확인하였다.

데이터프레임을 직접 확인하지 않고, 빈도수 배열을 생성할 수 있지만, 장소별 발화열원 항목이 존재하지 않을 수 있으므로, 직접 빈도수를 확인한 후 배열을 생성하였다.

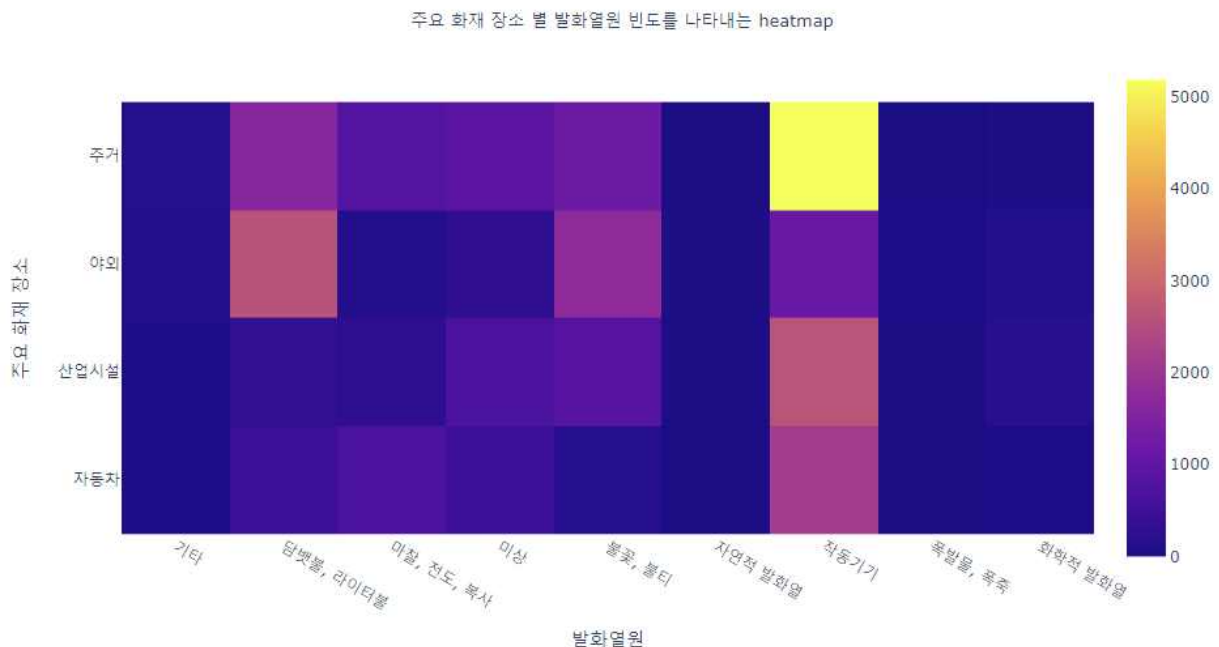
실제로 주거와 자동차에는 폭발물·폭죽에 대한 빈도수가 0으로 나타나지 않았다.

코드)

```
print(living_count)
print(outside_count)
print(industry_count)
print(vehicle_count)
```


3) 주요 화재 장소별 발화열원 빈도를 나타내는 heatmap

```
fig = px.imshow(data, labels=dict(x="발화열원", y="주요 화재 장소"),
                x=['기타', '담뱃불,ライター불', '마찰, 전도, 복사', '미상', '불꽃, 불티',
                  '자연적 발화열', '작동기기', '폭발물, 폭죽', '화학적 발화열'],
                y=['주거', '야외', '산업시설', '자동차'])
fig.update_layout(
    title_text="주요 화재 장소 별 발화열원 빈도를 나타내는 heatmap",
    title_font=dict(size=13),
    title_x=0.5
)
fig.show()
```



위의 heatmap은 주요 화재 장소(주거, 야외, 산업시설, 자동차)에 대한 발화열원의 빈도를 나타낸다. 주거 장소에서의 발화열원은 작동기기가 압도적으로 높았으며, 다음으로 담뱃불·라이터불, 불꽃·불티, 미상, 마찰·전도·복사 등의 순으로 빈도가 높았다. 다음으로 야외 장소에서는 담뱃불·라이터불, 불꽃·불티, 작동기기 순서로 빈도가 높았고, 나머지 열원들은 빈도수에 큰 차이가 없는 것으로 확인됐다. 마지막으로 산업시설과 자동차에서는 주거와 마찬가지로 작동기기가 발화열원이 되는 경우가 압도적으로 많았고, 나머지는 크게 차이가 없는 것으로 확인됐다. 주요 화재 장소에 따른 발화열원의 빈도 확인을 통해, 다수의 장소에서 작동기기가 발화열원으로 작용하는 경우가 많을 것으로 생각될 수 있다.

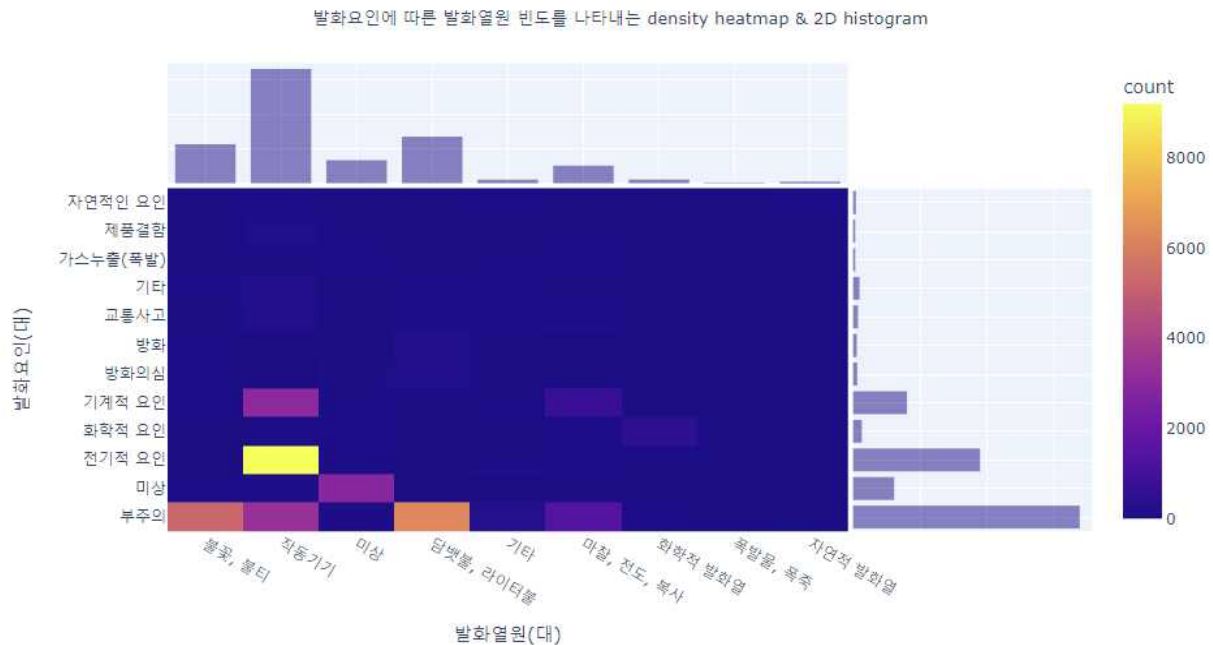
코드)

```
fig = px.imshow(data, labels=dict(x="발화열원", y="주요 화재 장소"),
                x=['기타', '담뱃불,ライター불', '마찰, 전도, 복사', '미상', '불꽃, 불티',
                  '자연적 발화열', '작동기기', '폭발물, 폭죽', '화학적 발화열'],
                y=['주거', '야외', '산업시설', '자동차'])
fig.update_layout(
    title_text="주요 화재 장소 별 발화열원 빈도를 나타내는 heatmap",
    title_font=dict(size=13),
    title_x=0.5
)
fig.show()
```

4.6 Density heat map & 2D-Histogram

1) 발화요인에 따른 발화열원 빈도를 나타내는 density heat map & 2D-histogram

```
fig = px.density_heatmap(df, x="발화열원(대)", y="발화요인(대)", marginal_x="histogram", marginal_y="histogram")
fig.update_layout(
    title_text="발화요인에 따른 발화열원 빈도를 나타내는 density heatmap & 2D histogram",
    title_font=dict(size=13),
    title_x=0.5
)
fig.show()
```



위의 density plot은 발화요인에 따른 발화열원의 빈도(밀도)를 나타낸다.

발화요인은 대부분 부주의, 미상, 전기적 요인, 화학적 요인으로 나타난 것으로 보이며, 이것은 density heat map뿐만 아니라 가로축의 발화요인에 대한 histogram으로도 확인할 수 있다.

그중에서도 부주의에 의한 화재 발생이 잦았으며, 담뱃불·라이터불, 불꽃·불티, 작동기기, 마찰·전도·복사의 순으로 자주 발화열원으로 밝혀진 것으로 보인다.

다음으로 많은 화재를 발생시킨 발화요인은 전기적 요인으로, 작동기기가 발화열원이 된 화재가 대다수임을 확인할 수 있다.

다음으로 많은 화재를 발생시킨 기계적 요인 또한, 작동기기가 발화열원이 된 사례가 대다수였고, 마지막으로 알려지지 않은 요인(미상)은 발화열원 또한 알려지지 않은 경우가 대부분임을 알 수 있다.

발화열원의 기준으로 histogram을 확인하면 발화요인을 특정하지 않을 때, 작동기기가 가장 빈번하게 발화열원으로 작용한다는 것을 쉽게 알 수 있다.

이처럼 부주의로 발생하는 화재가 가장 빈번하다는 사실을 통해, 적절한 조치나 교육을 통해 많은 화재를 줄일 수 있을 것으로 생각된다.

코드)

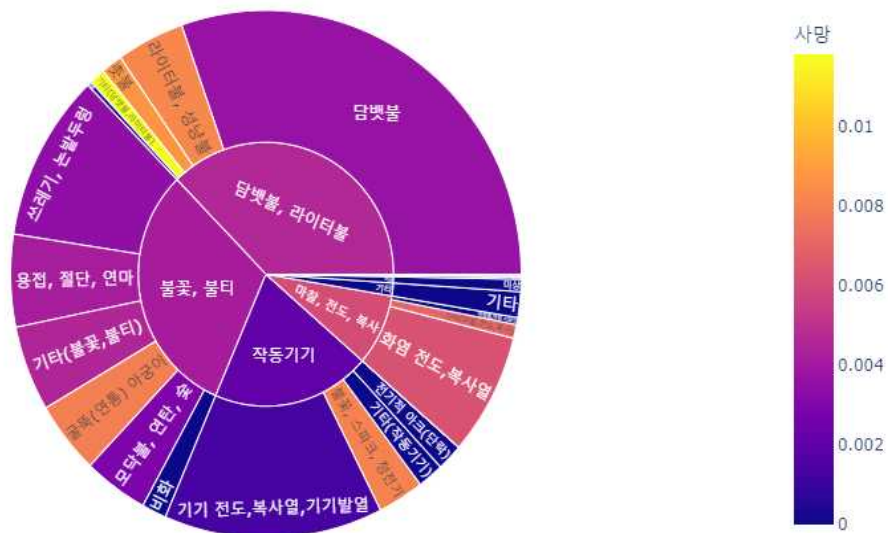
```
fig = px.density_heatmap(df, x="발화열원(대)", y="발화요인(대)", marginal_x="rug",
marginal_y="histogram")
fig.update_layout(
    title_text="발화요인에 따른 발화열원 빈도를 나타내는 density heatmap & 2D histogram",
    title_font=dict(size=13),
    title_x=0.5
)
fig.show()
```

4.7 Sunburst plot

1) 특정 발화요인(부주의)으로 발생하는 화재의 발화 열원의 빈도와 평균 사망자 수를 나타내는 sunburst plot

```
fire_factor_careless=df.loc[df['발화요인(대)']=='부주의']
fig = px.sunburst(fire_factor_careless, path=['발화열원(대)', '발화열원(소)'],
                  color='사망')
fig.update_layout(
    title_text='부주의로 발생하는 화재의 발화열원의 빈도를 나타내는 sunburst plot',
    title_font=dict(size=13),
    title_x=0.5
)
fig.show()
```

부주의로 발생하는 화재의 발화열원의 빈도를 나타내는 sunburst plot



대다수의 화재를 발생시키는 요인이 부주의라는 사실을 통해, 부주의로 발생하는 화재에 대해 알아보 고자 발화요인(=부주의), 발화열원, 사망 결럼을 이용해 sunburst plot을 나타냈다.

4.6)에서 확인하였듯, 부주의로 인한 화재의 발화열원은 주로 담뱃불·라이터불, 불꽃·불티, 작동기기, 마찰·전도·복사인 것으로 보인다.

자세히 확인하면, 담뱃불·라이터불의 하위 항목에 담뱃불이 많은 비중을 차지하는 것을 확인할 수 있 다. 그러나 평균 사망자 수는 비중이 가장 적은 기타 항목에서 약 0.01로 가장 높음을 확인할 수 있다. 다음으로 불꽃·불티의 하위 항목에선 쓰레기·논발두렁이 가장 많은 비중을 차지하는 것으로 나타났으 나, 나머지 항목들 또한 비등한 수준인 것을 확인할 수 있다. 추가로 평균 사망자 수는 굴뚝(연통) 아 궁이 항목이 약 0.008로 가장 높음을 확인할 수 있다.

다음으로 작동기기의 하위 항목에선 기기 전도·복사열·기기발열이 많은 비중을 차지하는 것으로 확인 된다. 평균 사망자 수는 다음으로 비중이 많은 불꽃·스파크·정전기가 약 0.008로 가장 높은 것으로 보 인다.

마지막으로 마찰·전도·복사의 하위 항목에선 화염 전도·복사열이 가장 많은 비중을 차지하며, 평균 사 망자 수는 기타(마찰·전도·복사)가 약 0.007로 가장 높음을 확인할 수 있다.

결과적으로 상위항목(발화열원(대))에서 마찰·전도·복사가 발화열원일 경우에, 평균 사망자 수가 약 0.006으로 가장 높음을 알 수 있었다. 그러나 부주의로 인한 모든 화재에서 평균 사망자 수가 매우 적 은 점을 통해, 화재로 인한 사망자 수를 줄이는 것이 목표라면 사망자 수가 많은 특정 화재를 확인하 여 특징을 파악하는 것이 적절하다고 생각된다.

코드)

```
fire_factor_careless=df.loc[df['발화요인(대)']=='부주의']
fig = px.sunburst(fire_factor_careless, path=['발화열원(대)', '발화열원(소)'], color='사망')
fig.update_layout(
    title_text='부주의로 발생하는 화재의 발화열원의 빈도를 나타내는 sunburst plot',
    title_font=dict(size=13),
    title_x=0.5
)
fig.show()
```

4.8 Tree map

1) 발화열원별 화재 수 및 평균 재산피해액에 대한 tree map

```
fig = px.treemap(df, path=[px.Constant('발화열원(대)'),
                                '발화열원(대)', '발화열원(소)'],
                color='재산피해')
fig.update_layout(
    title_text='발화열원별 화재 발생 횟수 및 재산피해액의 평균에 대한 tree map',
    title_font=dict(size=13),
    title_x=0.5
)
fig.show()
```

발화열원별 화재 발생 횟수 및 재산피해액의 평균에 대한 tree map



위의 시각화 방식은 발화열원별 발생한 화재 수에 비례하여 사각형의 크기로 나타내고, 평균 사망자를 비교하여 색으로 표현한 tree map이다. tree map은 계층적 구조이며, 위의 tree map에서 상위 항목은 '발화열원(대)'이며 하위 항목은 '발화열원(소)'이다.

4.5)에서 '발화열원(대)'별 화재 발생 횟수를 확인하였지만, tree map에선 '발화열원(대)'의 세부적인 하위 항목인 '발화열원(소)'별 화재 발생 횟수를 확인할 수 있을 뿐만 아니라 평균 재산피해 또한 비교할 수 있다는 점에서 차이가 있다.

'발화열원(대)'의 기준으로 작동기기, 담뱃불·라이터불, 불꽃·불티, 미상, 마찰·전도·복사 등의 순으로 많은 화재에서 발화열원으로 발견되었고, 대부분의 발화열원에서 비슷한 평균 재산피해를 나타내는 반면, 발화열원이 알려지지 않은 미상 항목에서 약 64122로 높은 평균 재산피해를 보이고 있다. 이외에도 화학적 발화열 항목도 약 36596로 높은 평균 재산피해를 나타내고 있다.

코드)

```
fig = px.treemap(df, path=[px.Constant('발화열원(대)'),
                                '발화열원(대)', '발화열원(소)'],
                color='재산피해')
fig.update_layout(
    title_text='발화열원별 화재 수 및 평균 재산피해액에 대한 tree map',
    title_font=dict(size=13),
    title_x=0.5
)
fig.show()
```