

```

120 def organize_data(self):
121     population = self.edit_column()
122     population['20-39세'] = population['20 - 24세'] + population['25 - 29세'] + \
123         population['30 - 34세'] + population['35 - 39세']
124
125     population['65세이상'] = population['65 - 69세'] + population['70 - 74세'] + \
126         population['75 - 79세'] + population['80 - 84세'] + \
127         population['85 - 89세'] + population['90 - 94세'] + \
128         population['95 - 99세'] + population['100+']
129
130     # pivot_table을 이용하여 광역시도, 시도를 index로 두고, 구분으로 세로로 첫 번째 컬럼을 잡고, value에 인구수, 20~39세, 65세이상으로 정리해 본다.
131     pop = pd.pivot_table(population,
132         index=['광역시도', '시도'],
133         columns=['구분'],
134         values=['인구수', '20-39세', '65세이상'])
135
136     # 소멸비율이라는 컬럼에 인구소멸위기지역을 계산하기 위한 식을 적용한다.
137     # 이 비율이 1보다 작으면 인구소멸위기지역으로 볼 수 있다.
138     pop['소멸비율'] = pop['20-39세', '여자'] / (pop['65세이상', '합계'] / 2)
139
140     # 소멸위기지역인지를 boolean으로 지정해 본다
141     pop['소멸위기지역'] = pop['소멸비율'] < 1.0
142
143     # pivot_table로 잘 정리가 된 상태에서 .reset_index로 pivot_table의 result 속성을 다시 설정한다.
144     pop.reset_index(inplace=True)
145
146     # 이중 column을 해제하기 위해 두 컬럼 제목을 합쳐 다시 지정한다.
147     tmp_columns = [pop.columns.get_level_values(0)[n] + \
148         pop.columns.get_level_values(1)[n]
149         for n in range(0, len(pop.columns.get_level_values(0)))]
150     pop.columns = tmp_columns
151     ic(pop.info())
152     return pop

```