



Chapter 6 - exercise 3: Year - Pop - gdp_cap - life_exp

Cho dữ liệu year, pop, gdp_cap, life_exp từ tập tin du_lieu_year_gap_pop_life.txt, col từ tập tin col.txt

Hãy thực hiện các yêu cầu sau:

1. Tạo year, pop. In item cuối của year và pop
2. Vẽ biểu đồ line của year và pop với x-axis: year, y-axis: pop
3. Tạo gdp_cap, life_exp. In item cuối của gdp_cap, life_exp
4. Vẽ biểu đồ line của gdp_cap, life_exp với x-axis: gdp_cap, y-axis: life_exp. Biểu đồ này có thể xem được không? Nếu không thì đề xuất một loại biểu đồ phù hợp
5. Vẽ histogram của life_exp, màu cột xanh, viền đỏ
6. Vẽ histogram với bins = 5, màu cột xanh dương, viền đỏ
7. Vẽ histogram với bins = 20, màu cột xanh dương, viền đỏ
8. Từ life_exp1950 trong tập tin => Tạo life_exp1950. Vẽ histogram life_exp1950, với bins = 15, màu cột xanh dương, viền đỏ
9. Tạo scatter plot của gdp_gap và life_exp, sử dụng plt.xscale('log'). Thiết lập xlabel, ylabel, title
10. Tạo Scatter plot của gdp_gap và life_exp, sử dụng plt.xscale('log'). Thiết lập xlabel, ylabel, title. plt.scatter(gdp_cap, life_exp). Với: tick_val = [1000, 10000, 100000] và tick_lab = ['1k', '10k', '100k'] => plt.xticks(tick_val, tick_lab)
11. Đưa pop vào array np_pop. Vẽ scatter plot của gdp_cap, life_exp, với s = np_pop * 2, màu magenta. Thiết lập xlabel, ylabel, title và plt.xticks([1000, 10000, 100000], ['1k', '10k', '100k'])
12. Tạo col từ col trong tập tin col.txt. Vẽ scatter plot của gdp_cap, life_exp, với s = np.array(pop) * 2, màu c = col, alpha=0.8. Thiết lập xlabel, ylabel, title và plt.xticks([1000, 10000, 100000], ['1k', '10k', '100k'])
13. Tạo col từ col trong tập tin col.txt. Vẽ scatter plot của gdp_cap, life_exp, với s = np.array(pop) * 2, màu c = col, alpha=0.8. Thiết lập xlabel, ylabel, title và plt.xticks([1000, 10000, 100000], ['1k', '10k', '100k']). Thêm text cho 2 nơi trên biểu đồ là India và China: plt.text(1550, 71, 'India'), plt.text(5700, 80, 'China'). Thêm lưới cho biểu đồ

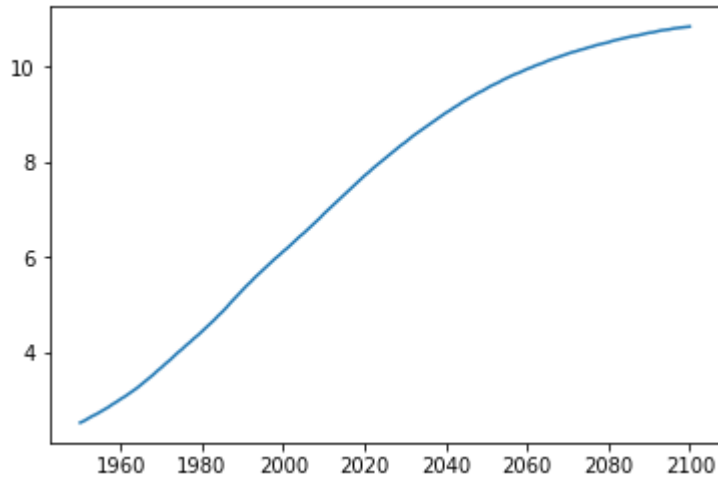
```
In [10]: # Câu 1:
year = [1950, 1951, 1952, 1953, 1954, 1955, 1956, 1957, 1958, 1959, 1960, 1961, 1962, 1963, 1964, 1965, 1966, 1967, 1968, 1969, 1970, 1971, 1972, 1973, 1974, 1975, 1976, 1977, 1978, 1979, 1980, 1981, 1982, 1983, 1984, 1985, 1986, 1987, 1988, 1989, 1990, 1991, 1992, 1993, 1994, 1995, 1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019]
```

```
In [11]: pop = [2.53, 2.57, 2.62, 2.67, 2.71, 2.76, 2.81, 2.86, 2.92, 2.97, 3.03, 3.08, 3.11, 3.15, 3.19, 3.23, 3.27, 3.31, 3.35, 3.39, 3.43, 3.47, 3.51, 3.55, 3.59, 3.63, 3.67, 3.71, 3.75, 3.79, 3.83, 3.87, 3.91, 3.95, 3.99, 4.03, 4.07, 4.11, 4.15, 4.19, 4.23, 4.27, 4.31, 4.35, 4.39, 4.43, 4.47, 4.51, 4.55, 4.59, 4.63, 4.67, 4.71, 4.75, 4.79, 4.83, 4.87, 4.91, 4.95, 4.99, 5.03, 5.07, 5.11, 5.15, 5.19, 5.23, 5.27, 5.31, 5.35, 5.39, 5.43, 5.47, 5.51, 5.55, 5.59, 5.63, 5.67, 5.71, 5.75, 5.79, 5.83, 5.87, 5.91, 5.95, 5.99, 6.03, 6.07, 6.11, 6.15, 6.19, 6.23, 6.27, 6.31, 6.35, 6.39, 6.43, 6.47, 6.51, 6.55, 6.59, 6.63, 6.67, 6.71, 6.75, 6.79, 6.83, 6.87, 6.91, 6.95, 6.99, 7.03, 7.07, 7.11, 7.15, 7.19, 7.23, 7.27, 7.31, 7.35, 7.39, 7.43, 7.47, 7.51, 7.55, 7.59, 7.63, 7.67, 7.71, 7.75, 7.79, 7.83, 7.87, 7.91, 7.95, 7.99, 8.03, 8.07, 8.11, 8.15, 8.19, 8.23, 8.27, 8.31, 8.35, 8.39, 8.43, 8.47, 8.51, 8.55, 8.59, 8.63, 8.67, 8.71, 8.75, 8.79, 8.83, 8.87, 8.91, 8.95, 8.99, 9.03, 9.07, 9.11, 9.15, 9.19, 9.23, 9.27, 9.31, 9.35, 9.39, 9.43, 9.47, 9.51, 9.55, 9.59, 9.63, 9.67, 9.71, 9.75, 9.79, 9.83, 9.87, 9.91, 9.95, 9.99, 10.03, 10.07, 10.11, 10.15, 10.19, 10.23, 10.27, 10.31, 10.35, 10.39, 10.43, 10.47, 10.51, 10.55, 10.59, 10.63, 10.67, 10.71, 10.75, 10.79, 10.83, 10.87, 10.91, 10.95, 10.99, 11.03, 11.07, 11.11, 11.15, 11.19, 11.23, 11.27, 11.31, 11.35, 11.39, 11.43, 11.47, 11.51, 11.55, 11.59, 11.63, 11.67, 11.71, 11.75, 11.79, 11.83, 11.87, 11.91, 11.95, 11.99, 12.03, 12.07, 12.11, 12.15, 12.19, 12.23, 12.27, 12.31, 12.35, 12.39, 12.43, 12.47, 12.51, 12.55, 12.59, 12.63, 12.67, 12.71, 12.75, 12.79, 12.83, 12.87, 12.91, 12.95, 12.99, 13.03, 13.07, 13.11, 13.15, 13.19, 13.23, 13.27, 13.31, 13.35, 13.39, 13.43, 13.47, 13.51, 13.55, 13.59, 13.63, 13.67, 13.71, 13.75, 13.79, 13.83, 13.87, 13.91, 13.95, 13.99, 14.03, 14.07, 14.11, 14.15, 14.19, 14.23, 14.27, 14.31, 14.35, 14.39, 14.43, 14.47, 14.51, 14.55, 14.59, 14.63, 14.67, 14.71, 14.75, 14.79, 14.83, 14.87, 14.91, 14.95, 14.99, 15.03, 15.07, 15.11, 15.15, 15.19, 15.23, 15.27, 15.31, 15.35, 15.39, 15.43, 15.47, 15.51, 15.55, 15.59, 15.63, 15.67, 15.71, 15.75, 15.79, 15.83, 15.87, 15.91, 15.95, 15.99, 16.03, 16.07, 16.11, 16.15, 16.19, 16.23, 16.27, 16.31, 16.35, 16.39, 16.43, 16.47, 16.51, 16.55, 16.59, 16.63, 16.67, 16.71, 16.75, 16.79, 16.83, 16.87, 16.91, 16.95, 16.99, 17.03, 17.07, 17.11, 17.15, 17.19, 17.23, 17.27, 17.31, 17.35, 17.39, 17.43, 17.47, 17.51, 17.55, 17.59, 17.63, 17.67, 17.71, 17.75, 17.79, 17.83, 17.87, 17.91, 17.95, 17.99, 18.03, 18.07, 18.11, 18.15, 18.19, 18.23, 18.27, 18.31, 18.35, 18.39, 18.43, 18.47, 18.51, 18.55, 18.59, 18.63, 18.67, 18.71, 18.75, 18.79, 18.83, 18.87, 18.91, 18.95, 18.99, 19.03, 19.07, 19.11, 19.15, 19.19, 19.23, 19.27, 19.31, 19.35, 19.39, 19.43, 19.47, 19.51, 19.55, 19.59, 19.63, 19.67, 19.71, 19.75, 19.79, 19.83, 19.87, 19.91, 19.95, 19.99, 20.03, 20.07, 20.11, 20.15, 20.19, 20.23, 20.27, 20.31, 20.35, 20.39, 20.43, 20.47, 20.51, 20.55, 20.59, 20.63, 20.67, 20.71, 20.75, 20.79, 20.83, 20.87, 20.91, 20.95, 20.99, 21.03, 21.07, 21.11, 21.15, 21.19, 21.23, 21.27, 21.31, 21.35, 21.39, 21.43, 21.47, 21.51, 21.55, 21.59, 21.63, 21.67, 21.71, 21.75, 21.79, 21.83, 21.87, 21.91, 21.95, 21.99, 22.03, 22.07, 22.11, 22.15, 22.19, 22.23, 22.27, 22.31, 22.35, 22.39, 22.43, 22.47, 22.51, 22.55, 22.59, 22.63, 22.67, 22.71, 22.75, 22.79, 22.83, 22.87, 22.91, 22.95, 22.99, 23.03, 23.07, 23.11, 23.15, 23.19, 23.23, 23.27, 23.31, 23.35, 23.39, 23.43, 23.47, 23.51, 23.55, 23.59, 23.63, 23.67, 23.71, 23.75, 23.79, 23.83, 23.87, 23.91, 23.95, 23.99, 24.03, 24.07, 24.11, 24.15, 24.19, 24.23, 24.27, 24.31, 24.35, 24.39, 24.43, 24.47, 24.51, 24.55, 24.59, 24.63, 24.67, 24.71, 24.75, 24.79, 24.83, 24.87, 24.91, 24.95, 24.99, 25.03, 25.07, 25.11, 25.15, 25.19, 25.23, 25.27, 25.31, 25.35, 25.39, 25.43, 25.47, 25.51, 25.55, 25.59, 25.63, 25.67, 25.71, 25.75, 25.79, 25.83, 25.87, 25.91, 25.95, 25.99, 26.03, 26.07, 26.11, 26.15, 26.19, 26.23, 26.27, 26.31, 26.35, 26.39, 26.43, 26.47, 26.51, 26.55, 26.59, 26.63, 26.67, 26.71, 26.75, 26.79, 26.83, 26.87, 26.91, 26.95, 26.99, 27.03, 27.07, 27.11, 27.15, 27.19, 27.23, 27.27, 27.31, 27.35, 27.39, 27.43, 27.47, 27.51, 27.55, 27.59, 27.63, 27.67, 27.71, 27.75, 27.79, 27.83, 27.87, 27.91, 27.95, 27.99, 28.03, 28.07, 28.11, 28.15, 28.19, 28.23, 28.27, 28.31, 28.35, 28.39, 28.43, 28.47, 28.51, 28.55, 28.59, 28.63, 28.67, 28.71, 28.75, 28.79, 28.83, 28.87, 28.91, 28.95, 28.99, 29.03, 29.07, 29.11, 29.15, 29.19, 29.23, 29.27, 29.31, 29.35, 29.39, 29.43, 29.47, 29.51, 29.55, 29.59, 29.63, 29.67, 29.71, 29.75, 29.79, 29.83, 29.87, 29.91, 29.95, 29.99, 30.03, 30.07, 30.11, 30.15, 30.19, 30.23, 30.27, 30.31, 30.35, 30.39, 30.43, 30.47, 30.51, 30.55, 30.59, 30.63, 30.67, 30.71, 30.75, 30.79, 30.83, 30.87, 30.91, 30.95, 30.99, 31.03, 31.07, 31.11, 31.15, 31.19, 31.23, 31.27, 31.31, 31.35, 31.39, 31.43, 31.47, 31.51, 31.55, 31.59, 31.63, 31.67, 31.71, 31.75, 31.79, 31.83, 31.87, 31.91, 31.95, 31.99, 32.03, 32.07, 32.11, 32.15, 32.19, 32.23, 32.27, 32.31, 32.35, 32.39, 32.43, 32.47, 32.51, 32.55, 32.59, 32.63, 32.67, 32.71, 32.75, 32.79, 32.83, 32.87, 32.91, 32.95, 32.99, 33.03, 33.07, 33.11, 33.15, 33.19, 33.23, 33.27, 33.31, 33.35, 33.39, 33.43, 33.47, 33.51, 33.55, 33.59, 33.63, 33.67, 33.71, 33.75, 33.79, 33.83, 33.87, 33.91, 33.95, 33.99, 34.03, 34.07, 34.11, 34.15, 34.19, 34.23, 34.27, 34.31, 34.35, 34.39, 34.43, 34.47, 34.51, 34.55, 34.59, 34.63, 34.67, 34.71, 34.75, 34.79, 34.83, 34.87, 34.91, 34.95, 34.99, 35.03, 35.07, 35.11, 35.15, 35.19, 35.23, 35.27, 35.31, 35.35, 35.39, 35.43, 35.47, 35.51, 35.55, 35.59, 35.63, 35.67, 35.71, 35.75, 35.79, 35.83, 35.87, 35.91, 35.95, 35.99, 36.03, 36.07, 36.11, 36.15, 36.19, 36.23, 36.27, 36.31, 36.35, 36.39, 36.43, 36.47, 36.51, 36.55, 36.59, 36.63, 36.67, 36.71, 36.75, 36.79, 36.83, 36.87, 36.91, 36.95, 36.99, 37.03, 37.07, 37.11, 37.15, 37.19, 37.23, 37.27, 37.31, 37.35, 37.39, 37.43, 37.47, 37.51, 37.55, 37.59, 37.63, 37.67, 37.71, 37.75, 37.79, 37.83, 37.87, 37.91, 37.95, 37.99, 38.03, 38.07, 38.11, 38.15, 38.19, 38.23, 38.27, 38.31, 38.35, 38.39, 38.43, 38.47, 38.51, 38.55, 38.59, 38.63, 38.67, 38.71, 38.75, 38.79, 38.83, 38.87, 38.91, 38.95, 38.99, 39.03, 39.07, 39.11, 39.15, 39.19, 39.23, 39.27, 39.31, 39.35, 39.39, 39.43, 39.47, 39.51, 39.55, 39.59, 39.63, 39.67, 39.71, 39.75, 39.79, 39.83, 39.87, 39.91, 39.95, 39.99, 40.03, 40.07, 40.11, 40.15, 40.19, 40.23, 40.27, 40.31, 40.35, 40.39, 40.43, 40.47, 40.51, 40.55, 40.59, 40.63, 40.67, 40.71, 40.75, 40.79, 40.83, 40.87, 40.91, 40.95, 40.99, 41.03, 41.07, 41.11, 41.15, 41.19, 41.23, 41.27, 41.31, 41.35, 41.39, 41.43, 41.47, 41.51, 41.55, 41.59, 41.63, 41.67, 41.71, 41.75, 41.79, 41.83, 41.87, 41.91, 41.95, 41.99, 42.03, 42.07, 42.11, 42.15, 42.19, 42.23, 42.27, 42.31, 42.35, 42.39, 42.43, 42.47, 42.51, 42.55, 42.59, 42.63, 42.67, 42.71, 42.75, 42.79, 42.83, 42.87, 42.91, 42.95, 42.99, 43.03, 43.07, 43.11, 43.15, 43.19, 43.23, 43.27, 43.31, 43.35, 43.39, 43.43, 43.47, 43.51, 43.55, 43.59, 43.63, 43.67, 43.71, 43.75, 43.79, 43.83, 43.87, 43.91, 43.95, 43.99, 44.03, 44.07, 44.11, 44.15, 44.19, 44.23, 44.27, 44.31, 44.35, 44.39, 44.43, 44.47, 44.51, 44.55, 44.59, 44.63, 44.67, 44.71, 44.75, 44.79, 44.83, 44.87, 44.91, 44.95, 44.99, 45.03, 45.07, 45.11, 45.15, 45.19, 45.23, 45.27, 45.31, 45.35, 45.39, 45.43, 45.47, 45.51, 45.55, 45.59, 45.63, 45.67, 45.71, 45.75, 45.79, 45.83, 45.87, 45.91, 45.95, 45.99, 46.03, 46.07, 46.11, 46.15, 46.19, 46.23, 46.27, 46.31, 46.35, 46.39, 46.43, 46.47, 46.51, 46.55, 46.59, 46.63, 46.67, 46.71, 46.75, 46.79, 46.83, 46.87, 46.91, 46.95, 46.99, 47.03, 47.07, 47.11, 47.15, 47.19, 47.23, 47.27, 47.31, 47.35, 47.39, 47.43, 47.47, 47.51, 47.55, 47.59, 47.63, 47.67, 47.71, 47.75, 47.79, 47.83, 47.87, 47.91, 47.95, 47.99, 48.03, 48.07, 48.11, 48.15, 48.19, 48.23, 48.27, 48.31, 48.35, 48.39, 48.43, 48.47, 48.51, 48.55, 48.59, 48.63, 48.67, 48.71, 48.75, 48.79, 48.83, 48.87, 48.91, 48.95, 48.99, 49.03, 49.07, 49.11, 49.15, 49.19, 49.23, 49.27, 49.31, 49.35, 49.39, 49.43, 49.47, 49.51, 49.55, 49.59, 49.63, 49.67, 49.71, 49.75, 49.79, 49.83, 49.87, 49.91, 49.95, 49.99, 50.03, 50.07, 50.11, 50.15, 50.19, 50.23, 50.27, 50.31, 50.35, 50.39, 50.43, 50.47, 50.51, 50.55, 50.59, 50.63, 50.67, 50.71, 50.75, 50.79, 50.83, 50.87, 50.91, 50.95, 50.99, 51.03, 51.07, 51.11, 51.15, 51.19, 51.23, 51.27, 51.31, 51.35, 51.39, 51.43, 51.47, 51.51, 51.55, 51.59, 51.63, 51.67, 51.71, 51.75, 51.79, 51.83, 51.87, 51.91, 51.95, 51.99, 52.03, 52.07, 52.11, 52.15, 52.19, 52.23, 52.27, 52.31, 52.35, 52.39, 52.43, 52.47, 52.51, 52.55, 52.59, 52.63, 52.67, 52.71, 52.75, 52.79, 52.83, 52.87, 52.91, 52.95, 52.99, 53.03, 53.07, 53.11, 53.15, 53.19, 53.23, 53.27, 53.31, 53.35, 53.39, 53.43, 53.47, 53.51, 53.55, 53.59, 53.63, 53.67, 53.71, 53.75, 53.79, 53.83, 53.87, 53.91, 53.95, 53.99, 54.03, 54.07, 54.11, 54.15, 54.19, 54.23, 54.27, 54.31, 54.35, 54.39, 54.43, 54.47, 54.51, 54.55, 54.59, 54.63, 54.67, 54.71, 54.75, 54.79, 54.83, 54.87, 54.91, 54.95, 54.99, 55.03, 55.07, 55.11, 55.15, 55.19, 55.23, 55.27, 55.31, 55.35, 55.39, 55.43, 55.47, 55.51, 55.55, 55.59, 55.63, 55.67, 55.71, 55.75, 55.79, 55.83, 55.87, 55.91, 55.95, 55.99, 56.03, 56.07, 56.11, 56.15, 56.19, 56.23, 56.27, 56.31, 56.35, 56.39, 56.43, 56.47, 56.51, 56.55, 56.59, 56.63, 56.67, 56.71, 56.75, 56.79, 56.83, 56.87, 56.91, 56.95, 56.99, 57.03, 57.07, 57.11, 57.15, 57.19, 57.23, 57.27, 57.31, 57.35, 57.39, 57.43, 57.47, 57.51, 57.55, 57.59, 57.63, 57.67, 57.71, 57.75, 57.79, 57.83, 57.87, 57.91, 57.95, 57.99, 58.03, 58.07, 58.11, 58.15, 58.19, 58.23, 58.27, 58.31, 58.35, 58.39, 58.43, 58.47, 58.51, 58.55, 58.59, 58.63, 58.67, 58.71, 58.75, 58.79, 58.83, 58.87, 58.91, 58.95, 58.99, 59.03, 59.07, 59.11, 59.15, 59.19, 59.23, 59.27, 59.31, 59.35, 59.39, 59.43, 59.47, 59.51, 59.55, 59.59, 59.63, 59.67, 59.71, 59.75, 59.79, 59.83, 59.87, 59.91, 59.95, 59.99, 60.03, 60.07, 60.11, 60.15, 60.19, 60.23, 60.27, 60.31, 60.35, 60.39, 60.43, 60.47, 60.51, 60.55, 60.59, 60.63, 60.67, 60.71, 60.75, 60.79, 60.83, 60.87, 60.91, 60.95, 60.99, 61.03, 61.07, 61.11, 61.15, 61.19, 61.23, 61.27, 61.31, 61.35, 61.39, 61.43, 61.47, 61.51, 61.55, 61.59, 61.63, 61.67, 61.71, 61.75, 61.79, 61.83, 61.87, 61.91, 61.95, 61.99, 62.03, 62.07, 62.11, 62.15, 62.19, 62.23, 62.27, 62.31, 62.35, 62.39, 62.43, 62.47, 62.51, 62.55, 62.59, 62.63, 62.67, 62.71, 62.75, 62.79, 62.83, 62.87, 62.91, 62.95, 62.99, 63.03, 63.07, 63.11, 63.15, 63.19, 63.23, 63.27, 63.31, 63.35, 63.39, 63.43, 63.47, 63.51, 63.55, 63.59, 63.63, 63.67, 63.71, 63.75, 63.79, 63.83, 63.87, 63.91, 63.95, 63.99, 64.03, 64.07, 64.11, 64.15, 64.19, 64.23, 64.27, 64.31, 64.35, 64.39, 64.43, 64.47, 64.51, 64.55, 64.59, 64.63, 64.67, 64.71, 64.75, 64.79, 64.83, 64.87, 64.91, 64.95, 64.99, 65.03, 65.07, 65.11, 65.15, 65.19, 65.23, 65.27, 65.31, 65.35, 65.39, 65.43, 65.47, 65.51, 65.55, 65.59, 65.6
```

```
In [13]: # Câu 2: Vẽ biểu đồ line của year và pop với x-axis: year, y-axis: pop
import matplotlib.pyplot as plt

# Make a line plot: year on the x-axis, pop on the y-axis
plt.plot(year, pop)

# Display the plot with plt.show()
plt.show()
```



```
In [14]: # Câu 3:
gdp_cap = [974.5803384, 5937.029525999998, 6223.367465, 4797.231267, 12779.37964,
```

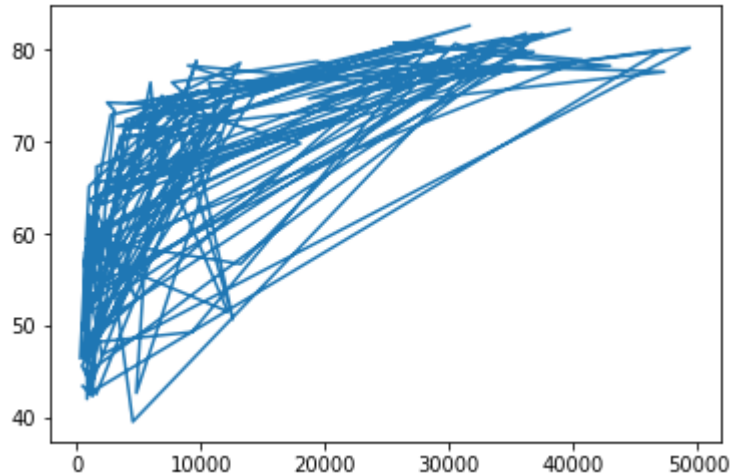
```
In [15]: life_exp = [43.828, 76.423, 72.301, 42.731, 75.32, 81.235, 79.829, 75.635, 64.062,
```

```
In [16]: # Print the last item of gdp_cap and life_exp
print(gdp_cap[-1])
print(life_exp[-1])
```

```
469.70929810000007
43.487
```

```
In [17]: # Câu 4: Vẽ biểu đồ line của gdp_cap, life_exp với x-axis: gdp_cap, y-axis: life_e
# Biểu đồ này có thể xem được không? Nếu không thì đề xuất một loại biểu đồ phù hợp
# Make a line plot, gdp_cap on the x-axis, life_exp on the y-axis
plt.plot(gdp_cap, life_exp)

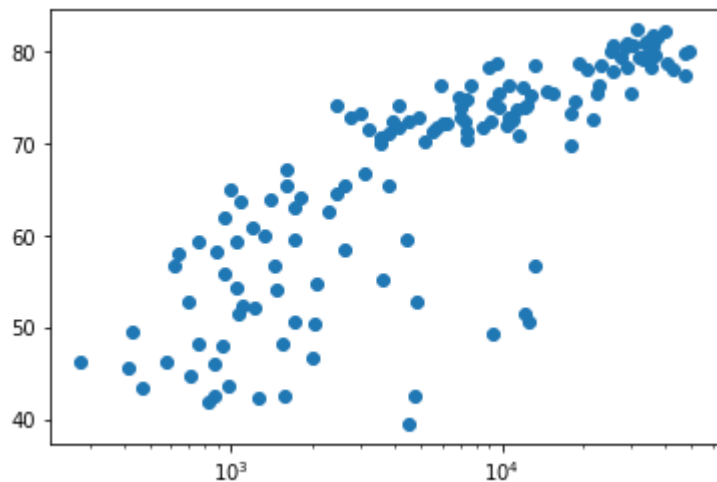
# Display the plot
plt.show()
```



```
In [18]: # Change the line plot below to a scatter plot
plt.scatter(gdp_cap, life_exp)

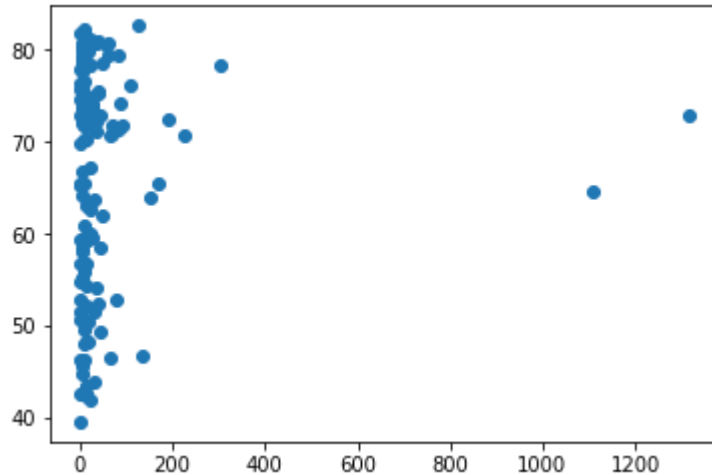
# Put the x-axis on a logarithmic scale
plt.xscale('log')

# Show plot
plt.show()
```



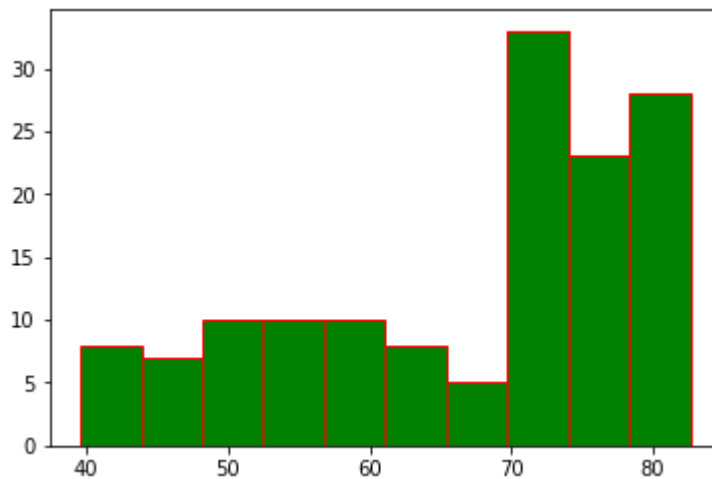
```
In [19]: pop = [31.889923, 3.600523, 33.333216, 12.420476, 40.301927, 20.434176, 8.199783,
# Build Scatter plot
plt.scatter(pop, life_exp)

# Show plot
plt.show()
```



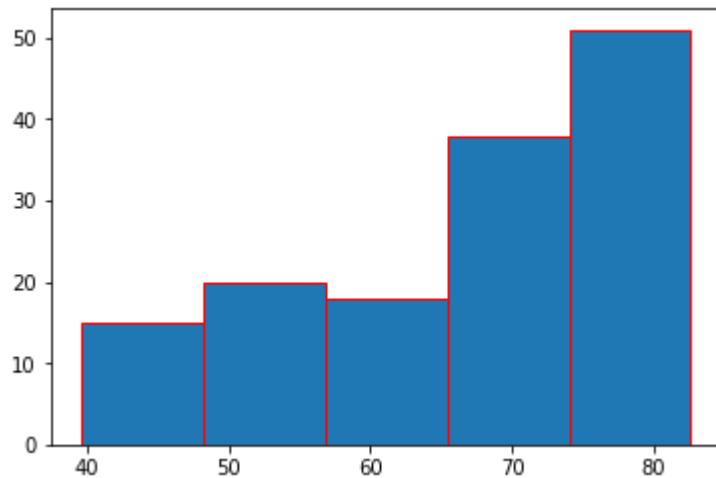
```
In [20]: # Câu 5: Vẽ histogram của life_exp, màu cột xanh, viền đỏ
# Create histogram of life_exp data
plt.hist(life_exp, color='g', edgecolor='r')

# Display histogram
plt.show()
```



```
In [21]: # Câu 6: Vẽ histogram với bins = 5, màu cột xanh dương, viền đỏ
plt.hist(life_exp, bins=5, edgecolor='r')

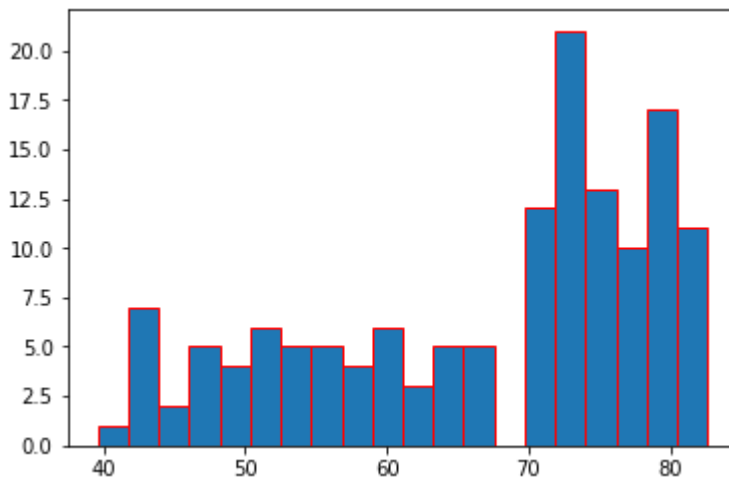
# Show and clean up plot
plt.show()
plt.clf()
```



<Figure size 432x288 with 0 Axes>

```
In [22]: # Câu 7: Vẽ histogram với bins = 20, màu cột xanh dương, viền đỏ
plt.hist(life_exp, bins=20, edgecolor='r')

# Show and clean up again
plt.show()
plt.clf()
```

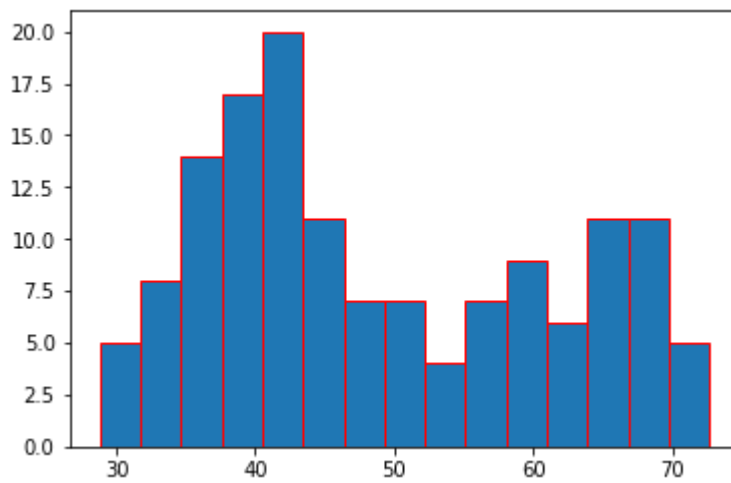


<Figure size 432x288 with 0 Axes>

```
In [23]: # Câu 8: Vẽ histogram life_exp1950, với bins = 15, màu cột xanh dương, viền đỏ
life_exp1950 = [28.8, 55.23, 43.08, 30.02, 62.48, 69.12, 66.8, 50.94, 37.48, 68.0,
```

```
In [24]: # Histogram of life_exp1950, 15 bins
plt.hist(life_exp1950, bins= 15, edgecolor='r')

# Show and clear plot again
plt.show()
plt.clf()
```



<Figure size 432x288 with 0 Axes>

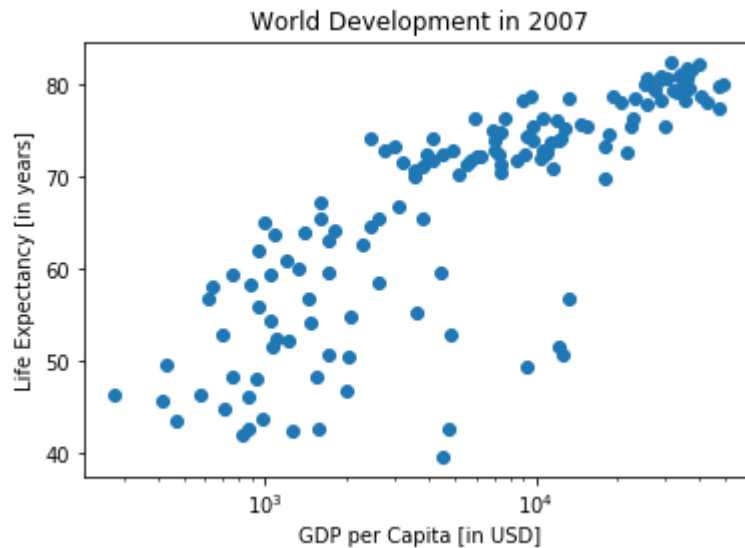
```
In [25]: # Câu 9: Tạo scatter plot của gdp_cap và life_exp, sử dụng plt.xscale('log'). Thiết
plt.scatter(gdp_cap, life_exp)
plt.xscale('log')

# Strings
xlab = 'GDP per Capita [in USD]'
ylab = 'Life Expectancy [in years]'
title = 'World Development in 2007'

# Add axis labels
plt.xlabel(xlab)
plt.ylabel(ylab)

# Add title
plt.title(title)

# After customizing, display the plot
plt.show()
```



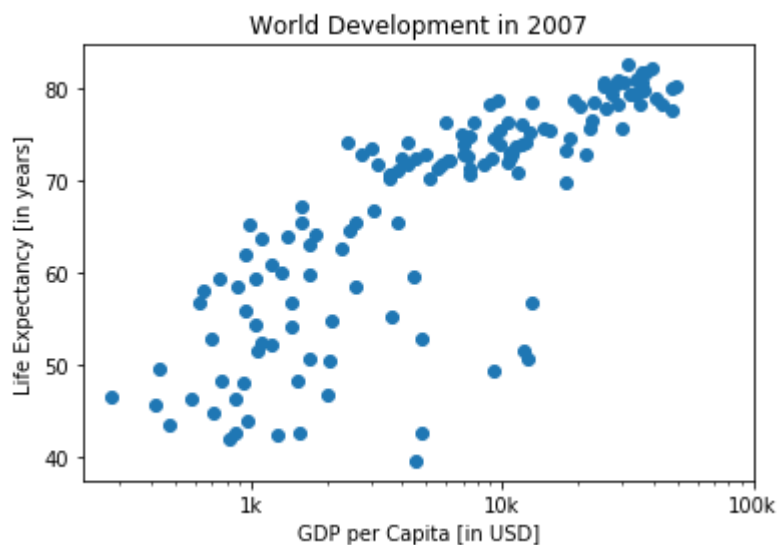
```
In [30]: ''' Câu 10: Tạo Scatter plot của gdp_cap và life_exp, sử dụng plt.xscale('log'). T
plt.scatter(gdp_cap, life_exp). Với:
tick_val = [1000,10000,100000]
tick_lab = ['1k','10k','100k']
plt.xticks(tick_val, tick_lab)
'''

plt.scatter(gdp_cap, life_exp)
# Previous customizations
plt.xscale('log')
plt.xlabel('GDP per Capita [in USD]')
plt.ylabel('Life Expectancy [in years]')
plt.title('World Development in 2007')

# Definition of tick_val and tick_lab
tick_val = [1000,10000,100000]
tick_lab = ['1k','10k','100k']

# Adapt the ticks on the x-axis
plt.xticks(tick_val, tick_lab)

# After customizing, display the plot
plt.show()
```




```
In [27]: '''
Câu 11: Đưa pop vào array np_pop
Vẽ scatter plot của gdp_cap, life_exp, với s = np_pop * 2, màu magenta
Thiết lập xlabel, ylabel, title và plt.xticks([1000, 10000, 100000],['1k', '10k',
'''

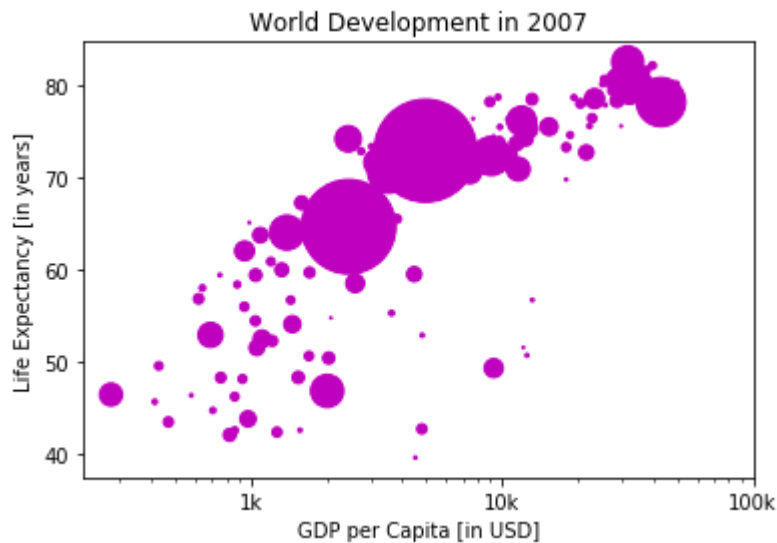
# Import numpy as np
import numpy as np

# Store pop as a numpy array: np_pop
np_pop = np.array(pop)

# Update: set s argument to np_pop
plt.scatter(gdp_cap, life_exp, s = np_pop * 2, color='m')

# Previous customizations
plt.xscale('log')
plt.xlabel('GDP per Capita [in USD]')
plt.ylabel('Life Expectancy [in years]')
plt.title('World Development in 2007')
plt.xticks([1000, 10000, 100000],['1k', '10k', '100k'])

# Display the plot
plt.show()
```

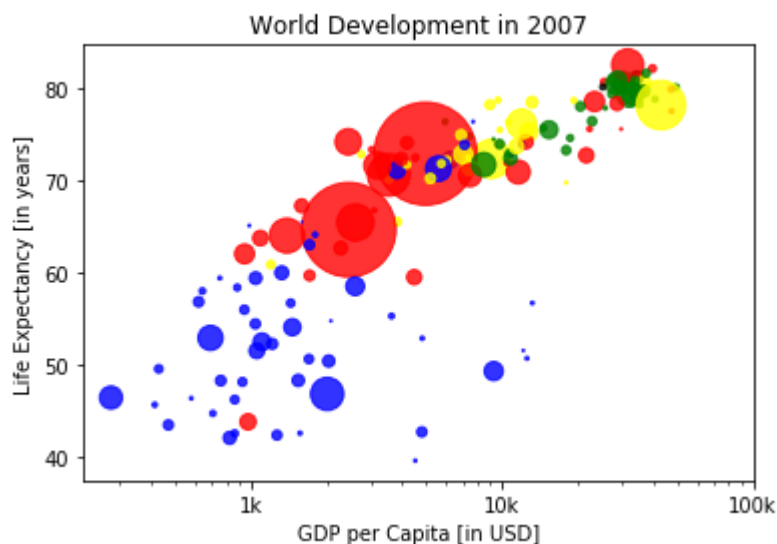


```
In [28]: '''
Câu 12: Tạo col từ col trong tập tin col.txt
Vẽ scatter plot của gdp_cap, life_exp, với s = np.array(pop) * 2, màu c = col, alp
Thiết lập xlabel, ylabel, title và plt.xticks([1000, 10000, 100000],['1k', '10k',
'''

col = ['red', 'green', 'blue', 'blue', 'yellow', 'black', 'green', 'red', 'red', '
      'yellow', 'green', 'blue', 'yellow', 'green', 'blue', 'blue', 'red', 'blue'
      'blue', 'yellow', 'red', 'yellow', 'blue', 'blue', 'blue', 'yellow', 'blue'
# Specify c and alpha inside plt.scatter()
plt.scatter(x = gdp_cap, y = life_exp, s = np.array(pop) * 2, c = col, alpha=0.8)

# Previous customizations
plt.xscale('log')
plt.xlabel('GDP per Capita [in USD]')
plt.ylabel('Life Expectancy [in years]')
plt.title('World Development in 2007')
plt.xticks([1000,10000,100000], ['1k', '10k', '100k'])

# Show the plot
plt.show()
```



```

In [29]: '''
Câu 13: Tạo col từ col trong tập tin col.txt
Vẽ scatter plot của gdp_cap, life_exp, với s = np.array(pop) * 2, màu c = col, alp
Thiết lập xlabel, ylabel, title và plt.xticks([1000, 10000, 100000],['1k', '10k',
Thêm text cho 2 nơi là India và China:
plt.text(1550, 71, 'India')
plt.text(5700, 80, 'China')
Thêm lưới cho biểu đồ
'''

plt.scatter(x = gdp_cap, y = life_exp, s = np.array(pop) * 2, c = col, alpha = 0.8

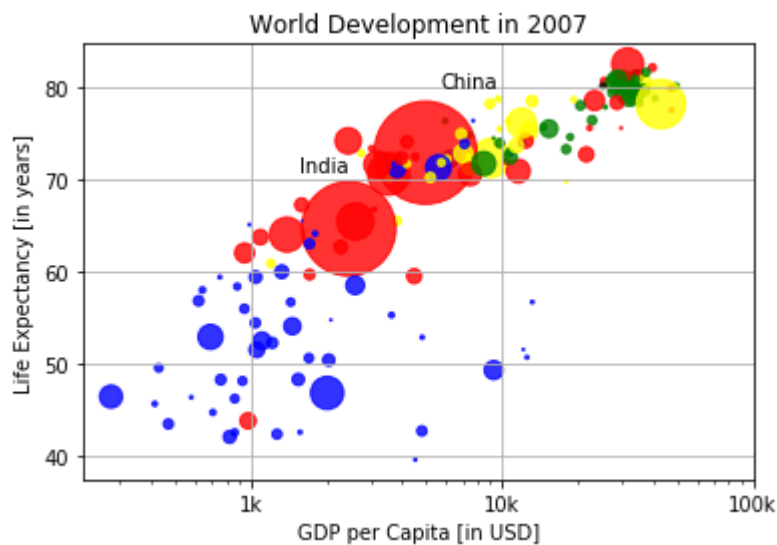
# Previous customizations
plt.xscale('log')
plt.xlabel('GDP per Capita [in USD]')
plt.ylabel('Life Expectancy [in years]')
plt.title('World Development in 2007')
plt.xticks([1000,10000,100000], ['1k', '10k', '100k'])

# Additional customizations
plt.text(1550, 71, 'India')
plt.text(5700, 80, 'China')

# Add grid() call
plt.grid()

# Show the plot
plt.show()

```



In []: