In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

In [2]:
```
from matplotlib import font_manager, rc # font_manager을 import
font_path = "C:/Windows/Fonts/H2HDRM.TTF" # font의 지정 경로
font = font_manager.FontProperties(fname=font_path).get_name()
rc('font', family=font)
```

In [3]:
```
file_path = '../Data/Income_per_all.csv'
file_path_1 = '../Data/Income_per1.csv'
file_path_10 = '../Data/Income_per10.csv'
Income = pd.read_csv(file_path)
Income_1 = pd.read_csv(file_path_1)
Income_10 = pd.read_csv(file_path_10)
```

In [4]:
```
Income.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 100 entries, 0 to 99
Data columns (total 4 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Percent        100 non-null    int64
 1   Quanity        100 non-null    int64
 2   Reference      100 non-null    int64
 3   Annual_Income  100 non-null    int64
dtypes: int64(4)
memory usage: 3.2 KB
```

In [28]:
```
Income.describe()
```

Out[28]:

	Percent	Quanity	Reference	Annual_Income
count	100.000000	100.000000	100.000000	100.00000
mean	50.500000	191672.730000	71753.130000	6177.55000
std	29.011492	0.446196	69539.883601	26836.19614
min	1.000000	191672.000000	199.000000	10.00000
25%	25.750000	191672.000000	31574.500000	1647.25000
50%	50.500000	191673.000000	53586.000000	2795.50000
75%	75.250000	191673.000000	91729.750000	4786.00000
max	100.000000	191673.000000	518366.000000	270446.00000

In [6]:
```
Income.head(10)
```

Out[6]:

	Percent	Quanity	Reference	Annual_Income

	Percent	Quanity	Reference	Annual_Income
0	1	191672	518366	270446
1	2	191673	268974	14033
2	3	191673	229513	11974
3	4	191672	207166	10808
4	5	191673	191091	9970
5	6	191673	179197	9349
6	7	191673	170135	8876
7	8	191672	162736	8490
8	9	191673	156169	8148
9	10	191673	150077	7830

In [7]:
```
Income[10:20]
```

Out[7]:

	Percent	Quanity	Reference	Annual_Income
10	11	191673	144447	7536
11	12	191672	139314	7268
12	13	191673	134575	7021
13	14	191673	130028	6784
14	15	191672	125660	6556
15	16	191673	121522	6340
16	17	191673	117564	6134
17	18	191673	114012	5948
18	19	191672	110567	5769
19	20	191673	107273	5597

In [8]:
```
Income[20:30]
```

Out[8]:

	Percent	Quanity	Reference	Annual_Income
20	21	191673	104192	5436
21	22	191673	101304	5285
22	23	191672	98551	5142
23	24	191673	95948	5006
24	25	191673	93454	4876
25	26	191672	91155	4756
26	27	191673	88894	4638
27	28	191673	86694	4523

	Percent	Quanity	Reference	Annual_Income
28	29	191673	84596	4414
29	30	191672	82533	4306

In [9]:
```
Income[30:40]
```

Out[9]:

	Percent	Quanity	Reference	Annual_Income
30	31	191673	80590	4205
31	32	191673	78749	4109
32	33	191673	76948	4015
33	34	191672	75204	3924
34	35	191673	73535	3836
35	36	191673	71907	3752
36	37	191673	70324	3669
37	38	191672	68935	3597
38	39	191673	67531	3523
39	40	191673	66160	3452

In [10]:
```
Income[40:50]
```

Out[10]:

	Percent	Quanity	Reference	Annual_Income
40	41	191672	64748	3378
41	42	191673	63413	3308
42	43	191673	62133	3242
43	44	191673	60897	3177
44	45	191672	59659	3113
45	46	191673	58448	3049
46	47	191673	57400	2995
47	48	191673	56312	2938
48	49	191672	55215	2881
49	50	191673	54131	2824

In [11]:
```
Income[50:60]
```

Out[11]:

	Percent	Quanity	Reference	Annual_Income
50	51	191673	53041	2767
51	52	191672	51985	2712

	Percent	Quanity	Reference	Annual_Income
52	53	191673	50932	2657
53	54	191673	49978	2607
54	55	191673	48953	2554
55	56	191672	48053	2507
56	57	191673	47164	2461
57	58	191673	46274	2414
58	59	191673	45731	2386
59	60	191672	44862	2341

In [12]:
```
Income[60:70]
```

Out[12]:

	Percent	Quanity	Reference	Annual_Income
60	61	191673	44011	2296
61	62	191673	43372	2263
62	63	191672	42588	2222
63	64	191673	41810	2181
64	65	191673	41271	2153
65	66	191673	40525	2114
66	67	191672	40203	2097
67	68	191673	39788	2076
68	69	191673	38785	2023
69	70	191673	37644	1964

In [13]:
```
Income_1.head(10)
```

Out[13]:

	Percent	Quanity	Reference	Annual_Income
0	0.1	19167	147132	76763
1	0.2	19167	65714	34285
2	0.3	19167	52032	27147
3	0.4	19168	44933	23442
4	0.5	19167	40407	21082
5	0.6	19167	37314	19468
6	0.7	19167	35019	18270
7	0.8	19168	33271	17358
8	0.9	19167	31848	16616
9	1.0	19167	30696	16015

In [14]:
```
Income_10.head(10)
```

Out[14]:

	Percent	Quanity	Reference	Annual_Income
0	10	1916727	2233424	35992
1	20	1916727	1244962	6495
2	30	1916727	927321	4838
3	40	1916727	729883	3808
4	50	1916727	604385	3153
5	60	1916727	496242	2588
6	70	1916727	417215	2176
7	80	1916727	324075	1690
8	90	1916727	210320	1097
9	100	1916727	80573	420

In [15]:
```
Income_10.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10 entries, 0 to 9
Data columns (total 4 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Percent        10 non-null     int64
 1   Quanity        10 non-null     int64
 2   Reference      10 non-null     int64
 3   Annual_Income  10 non-null     int64
dtypes: int64(4)
memory usage: 448.0 bytes
```

In [16]:
```
Income_Average = Income.groupby('Annual_Income').mean()
```

In [17]:
```
Income_Average
```

Out[17]:

Annual_Income	Percent	Quanity	Reference
10	100.0	191673.0	199.0
71	99.0	191673.0	1356.0
134	98.0	191673.0	2578.0
190	97.0	191672.0	3641.0
249	96.0	191673.0	4773.0
...	...	...	...
9970	5.0	191673.0	191091.0
10808	4.0	191672.0	207166.0

Annual_Income	Percent	Quanity	Reference
11974	3.0	191673.0	229513.0
14033	2.0	191673.0	268974.0
270446	1.0	191672.0	518366.0

100 rows × 3 columns

In [18]:
```
Income_10Inner = Income_Average[Income_Average.Percent <= 10]
```

In [19]:
```
Income_10Inner
```

Out[19]:

Annual_Income	Percent	Quanity	Reference
7830	10.0	191673.0	150077.0
8148	9.0	191673.0	156169.0
8490	8.0	191672.0	162736.0
8876	7.0	191673.0	170135.0
9349	6.0	191673.0	179197.0
9970	5.0	191673.0	191091.0
10808	4.0	191672.0	207166.0
11974	3.0	191673.0	229513.0
14033	2.0	191673.0	268974.0
270446	1.0	191672.0	518366.0

In [20]:
```
# 전체 평균 연봉
Total_Avg = Income.Annual_Income.mean()
```

In [21]:
```
Total_Avg
```

Out[21]: 6177.55

In [22]:
```
# 인구 백분위 50%의 연봉
```

In [23]:
```
Income[49:50]
```

Out[23]:

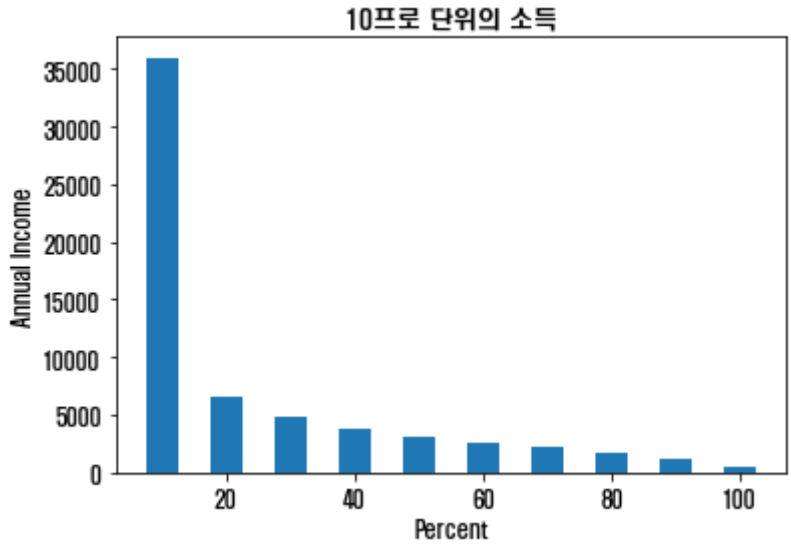
	Percent	Quanity	Reference	Annual_Income
49	50	191673	54131	2824

In [24]:
```
Per_Avg = Income.groupby('Percent')['Annual_Income'].mean()
```

In [25]:
```
Per_Avg.head(30)
```

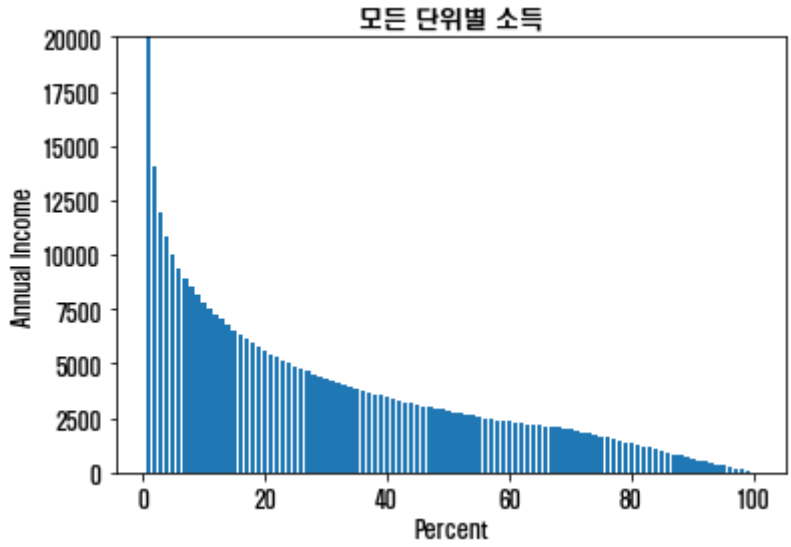
Out[25]:
```
Percent
1     270446.0
2      14033.0
3      11974.0
4      10808.0
5       9970.0
6       9349.0
7       8876.0
8       8490.0
9       8148.0
10      7830.0
11      7536.0
12      7268.0
13      7021.0
14      6784.0
15      6556.0
16      6340.0
17      6134.0
18      5948.0
19      5769.0
20      5597.0
21      5436.0
22      5285.0
23      5142.0
24      5006.0
25      4876.0
26      4756.0
27      4638.0
28      4523.0
29      4414.0
30      4306.0
Name: Annual_Income, dtype: float64
```

In [26]:
```
x = Income_10['Percent']
y = Income_10['Annual_Income']
plt.bar(x, y, width=5)
plt.xlabel('Percent')
plt.ylabel('Annual Income')
#plt.plot([3., 6], [Annual_Income, Annual_Income], "k-")
plt.title("10프로 단위의 소득")
plt.show()
```



In [27]:

```
x = Income['Percent']
y = Income['Annual_Income']
plt.bar(x, y)
plt.xlabel('Percent')
plt.ylabel('Annual Income')
plt.ylim([0, 20000])
plt.title("모든 단위별 소득")
plt.show()
```



In [ ]: