In this assignment, we are interested in exploring the effects of the exposome on pediatric asthma. Towards that we will assess and wrangle the provide data by answering the following questions. You will use the codebook, covariates, exposome and phenotype files. Submit your responses in a PDF file including your code and results. You can use Python or R. Do ask for help and if you have any questions.

1. Using the covariates dataset, group the data by sex "e3_sex_None". Which gender has the highest weight "hs_c_weight_None '' and what is its corresponding mean value? (20 Points)
   A. Males have the highest weight with a max of 71.1 and a mean of 29.0.

In [3]:
```
import pandas as pd

df_covariates = pd.read_csv('covariates.csv')
df_covariates.head()
```

Out[3]:

	ID	h_cohort	e3_sex_None	e3_yearbir_None	h_mbmi_None	hs_wgtgain_None	e3_gac_N
0	1	4	male	2008	25.510204	17.0	41.00
1	2	4	male	2007	26.491508	18.0	41.00
2	3	4	male	2008	30.116213	11.0	39.00
3	4	2	female	2005	21.048048	21.0	39.28
4	5	3	male	2005	22.151022	20.0	43.00

In [4]:
```
# Questoin 1.1
# group data based off sex
grouped = df_covariates.groupby('e3_sex_None')

# get max weight for males and females
weight_max = grouped['hs_c_weight_None'].max()
display(weight_max)

# get mean weight for males and females
weight_mean = grouped['hs_c_weight_None'].mean()
display(weight_mean)
```

```
e3_sex_None
female    69.9
male      71.1
Name: hs_c_weight_None, dtype: float64
e3_sex_None
female    27.936513
male      29.026696
Name: hs_c_weight_None, dtype: float64
```

2. Create a dataframe called “phenotype” using the phenotype dataset and answer the following questions:

   A. Count the number of missing values per column in the above dataframe.(20 Points)

   B. Extract and Print all individuals (IDs) from the dataframe what have a Body mass index categories at 6-11 years old - WHO reference of Overweight or Obese. Refer the Codebook file to find the field and values. (20 Points)

   C. Identify any records (IDs) that are duplicative in the records you extracted in Step b. (20 Points)

In [5]:
```
df_phenotype = pd.read_csv('phenotype.csv')
df_phenotype.head()
```

Out[5]:

	ID	e3_bw	hs_asthma	hs_zbmi_who	hs_correct_raven	hs_Gen_Tot	hs_bmi_c_cat
0	1	4100	0	0.30	18	84.0	2
1	2	4158	0	0.41	25	39.0	2
2	3	4110	1	3.33	13	40.0	4
3	4	3270	0	-0.76	28	54.5	2
4	5	3950	0	0.98	19	18.0	2

In [6]:
```
# Question 2.1
# sum null values for each column
nan_count = df_phenotype.isnull().sum()
print("Sum of null values for each column:")
print(nan_count)
```

```
Sum of null values for each column:
ID                  0
e3_bw               0
hs_asthma           0
hs_zbmi_who         0
hs_correct_raven    0
hs_Gen_Tot          0
hs_bmi_c_cat        0
dtype: int64
```

In [7]:
```
# Question 2.2
df_codebook = pd.read_csv('codebook.csv')

# get variable names for phenotype data
p_columns = df_phenotype.columns.tolist()
p_columns.remove('ID')

# extract relavent variables from codebook
codes = df_codebook[df_codebook['variable_name'].isin(p_columns)]
```

```python
# Looks like we want to use the hs_bmi_c_cat variable for question 2.2. and values

# get overweight and obese ids from phenotype data
overweight_id = df_phenotype['ID'][df_phenotype['hs_bmi_c_cat'] >= 3].tolist()
print("Overweight IDs:")
print(overweight_id)
print()

# Question 2.3
# Identify duplicate IDs
duplicates_mask = df_phenotype['ID'].duplicated(keep=False)
id_duplicate = df_phenotype['ID'][duplicates_mask].tolist()
print("Duplicate IDs:")
print(id_duplicate)
```

```
Overweight IDs:
[3, 10, 11, 12, 15, 16, 21, 25, 27, 30, 34, 39, 41, 43, 52, 54, 63, 67, 70, 75, 76,
77, 78, 79, 80, 81, 83, 84, 85, 87, 88, 95, 110, 112, 120, 121, 127, 140, 141, 142,
143, 144, 147, 150, 155, 159, 160, 170, 172, 174, 181, 183, 192, 193, 199, 200, 204,
206, 209, 213, 214, 225, 227, 229, 233, 247, 257, 259, 262, 266, 268, 270, 279, 284,
285, 292, 295, 297, 298, 299, 301, 302, 303, 304, 306, 309, 312, 315, 323, 324, 325,
328, 333, 336, 339, 340, 344, 345, 346, 351, 352, 359, 361, 362, 364, 366, 367, 379,
381, 385, 386, 388, 392, 397, 404, 407, 409, 410, 421, 424, 431, 433, 436, 437, 438,
441, 442, 443, 444, 447, 448, 452, 453, 458, 459, 461, 466, 469, 475, 481, 484, 490,
494, 497, 501, 502, 510, 512, 522, 523, 526, 530, 536, 540, 547, 550, 551, 552, 553,
554, 559, 562, 565, 574, 575, 577, 582, 584, 598, 599, 608, 609, 612, 615, 616, 617,
618, 620, 622, 623, 629, 631, 637, 638, 641, 645, 647, 648, 663, 664, 666, 670, 671,
680, 684, 686, 688, 690, 691, 702, 704, 707, 710, 712, 714, 717, 718, 719, 720, 721,
732, 735, 737, 740, 746, 747, 751, 756, 758, 759, 765, 766, 769, 784, 785, 790, 797,
798, 799, 800, 806, 809, 810, 815, 819, 822, 823, 829, 837, 840, 845, 846, 847, 848,
850, 852, 853, 855, 857, 861, 865, 868, 869, 870, 872, 875, 877, 880, 881, 886, 892,
893, 904, 906, 907, 908, 911, 914, 918, 920, 922, 924, 931, 936, 939, 941, 950, 957,
962, 965, 967, 969, 977, 984, 985, 987, 990, 991, 992, 993, 994, 998, 999, 1006, 100
8, 1010, 1011, 1012, 1014, 1016, 1019, 1020, 1027, 1029, 1030, 1032, 1041, 1043, 104
4, 1045, 1047, 1048, 1055, 1056, 1058, 1065, 1066, 1067, 1069, 1071, 1072, 1074, 107
5, 1078, 1081, 1085, 1089, 1090, 1092, 1093, 1096, 1099, 1101, 1103, 1106, 1108, 111
4, 1116, 1124, 1127, 1128, 1129, 1130, 1131, 1137, 1139, 1140, 1151, 1154, 1156, 115
7, 1161, 1166, 1167, 1169, 1175, 1177, 1181, 1182, 1183, 1190, 1191, 1192, 1196, 120
1, 1204, 1211, 1212, 1221, 1224, 1225, 1237, 1249, 1250, 1255, 1259, 1275, 1276, 128
5, 1290, 1291, 1295, 1297, 1299]

Duplicate IDs:
[]
```