## Лабораторная работа № 1

**Разведочный анализ данных. Исследование и визуализация данных.**

Выполнил: Ханмурзин Тагир ИУ5-64

In [1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(style="ticks")
```

***1. Текстовое описание выбранного Вами набора данных***

Датасет, который содержит 200 строк основной информации (идентификатор, возраст, пол, доход, счет расходов) о 200 клиентах соотвественно.

***2. Основные характеристики датасета***

In [12]:
```
data = pd.read_csv('Mall_Customers.csv', sep=",")
```

In [13]:
```
data.head() # Выводим первые 5 строк
```

Out[13]:

| | CustomerID | Gender | Age | Annual Income (k$) | Spending Score (1-100) |
|---|---|---|---|---|---|
| **0** | 1 | Male | 19 | 15 | 39 |
| **1** | 2 | Male | 21 | 15 | 81 |
| **2** | 3 | Female | 20 | 16 | 6 |
| **3** | 4 | Female | 23 | 16 | 77 |
| **4** | 5 | Female | 31 | 17 | 40 |

In [14]:
```
data.shape # Узнаём размер датасета
```

Out[14]:
```
(200, 5)
```

In [15]:
```
total_count = data.shape[0]
print('Всего строк: {}'.format(total_count))
```

```
Всего строк: 200
```

In [16]:
```
data.columns # Список колонок
```

Out[16]:
```
Index(['CustomerID', 'Gender', 'Age', 'Annual Income (k$)',
       'Spending Score (1-100)'],
      dtype='object')
```

In [17]:
```
data.dtypes # Список колонок с типами данных
```

Out[17]:
```
CustomerID                int64
Gender                   object
Age                       int64
Annual Income (k$)        int64
Spending Score (1-100)    int64
dtype: object
```

In [22]:
```
# Проверим наличие пустых значений
# Цикл по колонкам датасета
for col in data.columns:
    # Количество пустых значений - все значения заполнены
    temp_null_count = data[data[col].isnull()].shape[0]
    print('{} - {}'.format(col, temp_null_count))
```

```
CustomerID - 0
Gender - 0
Age - 0
Annual Income (k$) - 0
Spending Score (1-100) - 0
```

In [19]:
```
# Основные статистические характеристки набора данных
data.describe()
```

Out[19]:

| | CustomerID | Age | Annual Income (k$) | Spending Score (1-100) |
|---|---|---|---|---|
| **count** | 200.000000 | 200.000000 | 200.000000 | 200.000000 |
| **mean** | 100.500000 | 38.850000 | 60.560000 | 50.200000 |
| **std** | 57.879185 | 13.969007 | 26.264721 | 25.823522 |
| **min** | 1.000000 | 18.000000 | 15.000000 | 1.000000 |
| **25%** | 50.750000 | 28.750000 | 41.500000 | 34.750000 |
| **50%** | 100.500000 | 36.000000 | 61.500000 | 50.000000 |
| **75%** | 150.250000 | 49.000000 | 78.000000 | 73.000000 |
| **max** | 200.000000 | 70.000000 | 137.000000 | 99.000000 |

***3. Визуальное исследование датасета***

***4. Информация о корреляции признаков***

In [20]:
```
sns.heatmap(data.corr(), annot=True, fmt='.3f')
```

Out[20]:
```
<matplotlib.axes._subplots.AxesSubplot at 0xb1d6668>
```