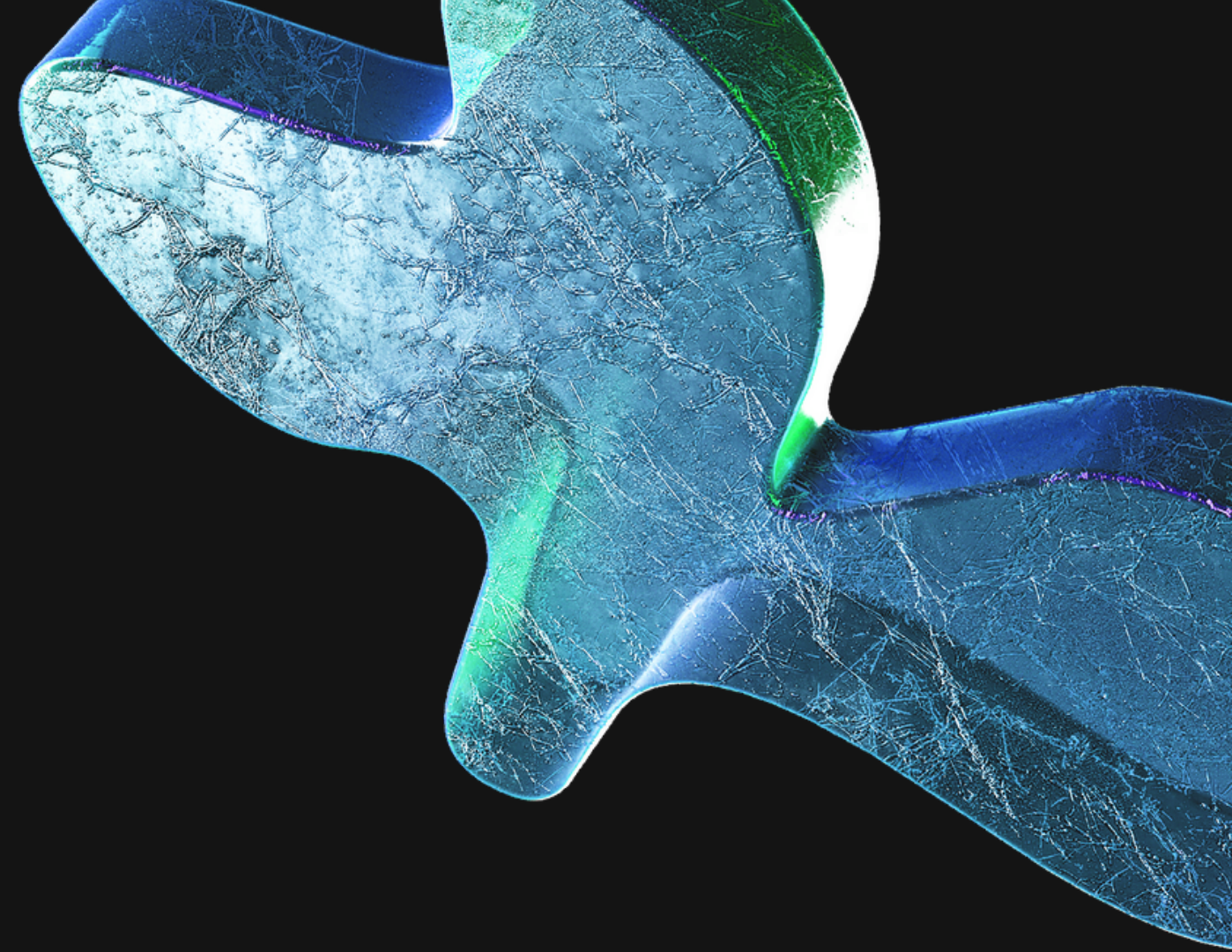


Кейс № 1. Результаты
экзаменов

Статистика

Статистика учебных учреждений и процесс разработки



Начало работы

Импорт необходимых модулей / Загрузка данных

```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3
4 df = pd.read_csv('StudentsPerformance .csv')
5
6 female_scores = df[df['gender'] == 'female'].groupby('parental level of education')[['math score', 'reading score', 'writing score']].mean()
7 male_scores = df[df['gender'] == 'male'].groupby('parental level of education')[['math score', 'reading score', 'writing score']].mean()
8
```

Импортируем и загружаем данные с помощью библиотеки pandas.

Вычисляем средние значения для каждого пола и предмета.

При выводе получаем среднее значение для каждого пола и предмета.

Средние значения оценок девочек:

	math score	reading score	writing score
parental level of education			
associate's degree	65.25	74.12	74.00
bachelor's degree	68.35	77.29	78.38
high school	59.35	68.20	66.69
master's degree	66.50	76.81	77.64
some college	65.41	73.55	74.05
some high school	59.30	69.11	68.29

Средние значения оценок мальчиков:

	math score	reading score	writing score
parental level of education			
associate's degree	70.76	67.43	65.41
bachelor's degree	70.58	68.09	67.65
high school	64.71	61.48	58.54
master's degree	74.83	73.13	72.61
some college	69.01	64.99	63.15
some high school	67.84	64.69	61.38

Следующий шаг

Вычисление средних значений для каждого типа обеда и предмета

```
11
12 standard_scores = df[df['lunch'] == 'standard'].groupby('parental level of education')[['math score', 'reading score', 'writing score']].mean()
13 free_scores = df[df['lunch'] == 'free/reduced'].groupby('parental level of education')[['math score', 'reading score', 'writing score']].mean()
14
15 print('Средние значения оценок для обеда "стандарт":\n', standard_scores.round(2))
16 print('Средние значения оценок для бесплатного обеда:\n', free_scores.round(2))
```

При выводе получаем среднее значение для платного и бесплатного обеда и предмета.

	math score	reading score	writing score
parental level of education			
associate's degree	70.68	72.67	71.95
bachelor's degree	73.19	75.42	75.85
high school	66.37	67.20	65.25
master's degree	75.63	78.86	79.66
some college	70.89	71.82	71.60
some high school	68.44	70.47	68.64
Средние значения оценок для бесплатного обеда:			
	math score	reading score	writing score
parental level of education			
associate's degree	62.61	67.65	66.03
bachelor's degree	63.00	68.93	69.23
high school	54.51	60.21	57.40
master's degree	61.17	70.29	69.88
some college	60.13	65.08	63.71
some high school	53.93	60.11	57.62

Построение графиков для средних значений оценок

Построение графиков для средних значений оценок каждого пола и предмета

```
fig, axes = plt.subplots(nrows=2, ncols=3, figsize=(12,8))
```

Эта строка создает фигуру с двумя строками и тремя столбцами вложенных графиков и устанавливает размер фигуры 12 дюймов в ширину и 8 дюймов в высоту.

Переменная `fig` представляет фигуру целиком, в то время как переменная `axes` представляет собой двумерный массив объектов `AxesSubplot`, представляющих каждый вложенный график.

```
for i, (gender, scores) in enumerate(zip(['Девочки', 'Мальчики'], [female_scores, male_scores])):
```

В этой строке выполняется итерация по каждому полу (женский и мужской) и их соответствующим средним баллам с использованием функции `zip()` для объединения двух списков в пары. Функция `enumerate()` добавляет переменную счетчика `i` для отслеживания текущей итерации.

Построение графиков для средних значений оценок

Построение графиков для средних значений оценок каждого пола и предмета

Далее выполняется итерация по каждому предмету (математика, чтение, письмо) и их соответствующему цвету (красный, зеленый, синий) с использованием функции `zip()` для объединения двух списков в пары. Функция `enumerate()` добавляет переменную счетчика `j` для отслеживания текущей итерации.

```
for i, (gender, scores) in enumerate(zip(['Девочки', 'Мальчики'], [female_scores, male_scores])):
    for j, (subject, color) in enumerate(zip(['math score', 'reading score', 'writing score'], ['r', 'g', 'b'])):
```


Построение графиков для средних значений оценок

Построение графиков для средних значений оценок каждого пола и предмета

Эта линия отображает горизонтальную столбчатую диаграмму для средних оценок текущего пола и предмета на текущем подзаголовке оси [i,j]

```
for j, (subject, color) in enumerate(zip(['math score', 'reading score', 'writing score'], ['r', 'g', 'b'])):
    ax = scores[subject].plot(kind='barh', color=color, ax=axes[i,j])
    ax.set_xlabel(subject)
    ax.set_ylabel('Уровень образования родителей')
    ax.set_title('{} - {} scores'.format(gender.capitalize(), subject.capitalize()))
```

Наконец, код отображает средние баллы для каждого пола и предмета с использованием горизонтальной столбчатой диаграммы. Функция `subplots` используется для создания рисунка с двумя строками и тремя столбцами, а функция `enumerate` используется для перебора пар `gender` и `subject`. Для каждой пары создается новый вспомогательный график с использованием метода `plot` объекта `DataFrame`. Методы `set_xlabel`, `set_ylabel` и `set_title` используются для добавления меток к подзаголовку. Функция `tight_layout` используется для настройки расположения вложенных графиков, а метод `show` используется для отображения диаграммы.

Построение графиков для средних значений оценок

Построение графиков для средних значений оценок каждого пола и предмета

И наконец выводим всё это чудо с помощью данных команд.

```
plt.tight_layout()
plt.show()
```

```
1  import pandas as pd
2  import matplotlib.pyplot as plt
3
4  df = pd.read_csv('StudentsPerformance .csv')
5
6  female_scores = df[df['gender'] == 'female'].groupby('parental level of education')[['math score', 'reading score', 'writing score']].mean()
7  male_scores = df[df['gender'] == 'male'].groupby('parental level of education')[['math score', 'reading score', 'writing score']].mean()
8
9  print('Средние значения оценок девочек:\n', female_scores.round(2))
10 print('Средние значения оценок мальчиков:\n', male_scores.round(2))
11
12 standard_scores = df[df['lunch'] == 'standard'].groupby('parental level of education')[['math score', 'reading score', 'writing score']].mean()
13 free_scores = df[df['lunch'] == 'free/reduced'].groupby('parental level of education')[['math score', 'reading score', 'writing score']].mean()
14
15 print('Средние значения оценок для обеда "стандарт":\n', standard_scores.round(2))
16 print('Средние значения оценок для бесплатного обеда:\n', free_scores.round(2))
17
18 fig, axes = plt.subplots(nrows=2, ncols=3, figsize=(12,8))
19 for i, (gender, scores) in enumerate(zip(['Девочки', 'Мальчики'], [female_scores, male_scores])):
20     for j, (subject, color) in enumerate(zip(['math score', 'reading score', 'writing score'], ['r', 'g', 'b'])):
21         ax = scores[subject].plot(kind='barh', color=color, ax=axes[i,j])
22         ax.set_xlabel(subject)
23         ax.set_ylabel('Уровень образования родителей')
24         ax.set_title('{} - {} scores'.format(gender.capitalize(), subject.capitalize()))
25
26 plt.tight_layout()
27 plt.show()
```

Весь код уложился мне за 27 (20) строк.

Результат сей чудо

Вывод данных

