

Первая часть

Задания этой части будут проверены в тренажёре автоматически.

Первая часть

Шаг 1. Откройте таблицу и изучите общую информацию о данных

Задание 1. Импортируйте библиотеку `pandas`. Считайте данные из csv-файла в датафрейм и сохраните в переменную `data`. Путь к файлу: `/datasets/data.csv`.

[Скачать датасет](#)

Задание 2. Выведите первые 20 строчек датафрейма `data` на экран.

Задание 3. Выведите основную информацию о датафрейме с помощью метода `info()`.

Шаг 2. Предобработка данных

Задание 4. Выведите количество пропущенных значений для каждого столбца. Используйте комбинацию двух методов.

Задание 5. В двух столбцах есть пропущенные значения. Один из них — `days_employed`. Пропуски в этом столбце вы обработаете на следующем этапе. Другой столбец с пропущенными значениями — `total_income` — хранит данные о доходах. На сумму дохода сильнее всего влияет тип занятости, поэтому заполнить пропуски в этом столбце нужно медианным значением по каждому типу из столбца `income_type`. Например, у человека с типом занятости сотрудник пропуск в столбце `total_income` должен быть заполнен медианным доходом среди всех записей с тем же типом.

Задание 6. В данных могут встречаться артефакты (аномалии) — значения, которые не отражают действительность и появились по какой-то ошибке. Таким артефактом будет отрицательное количество дней трудового стажа в столбце `days_employed`. Для реальных данных это нормально. Обработайте значения в этом столбце: замените все отрицательные значения положительными с помощью метода `abs()`.

Задание 7. Для каждого типа занятости выведите медианное значение трудового стажа в днях из столбца `days_employed`. У двух типов (безработные и пенсионеры) получатся аномально большие значения. Исправить такие значения сложно, поэтому оставьте их как есть.

Задание 8. Выведите перечень уникальных значений столбца `children`.

Задание 9. В столбце `children` есть два аномальных значения. Удалите строки, в которых встречаются такие аномальные значения из датафрейма `data`.

Задание 10. Ещё раз выведите перечень уникальных значений столбца `children`, чтобы убедиться, что артефакты удалены.

Задание 11. Заполните пропуски в столбце `days_employed` медианными значениями по каждому типу занятости `income_type`.

Задание 12. Убедитесь, что все пропуски заполнены. Проверьте себя и ещё раз выведите количество пропущенных значений для каждого столбца с помощью двух методов.

Задание 13. Замените вещественный тип данных в столбце `total_income` на целочисленный с помощью метода `astype()`.

Задание 14. Обработайте неявные дубликаты в столбце `education`. В этом столбце есть одни и те же значения, но записанные по-разному: с использованием заглавных и строчных букв. Приведите их к нижнему регистру.

Задание 15. Выведите на экран количество строк-дубликатов в данных. Если такие строки присутствуют, удалите их. Сбрасывать индексы после удаления строк дубликатов с помощью `reset_index(drop=True)` здесь не требуется.

Задание 16. На основании диапазонов, указанных ниже, создайте в датафрейме `data` столбец `total_income_category` с категориями:

0–30000 — 'E';
30001–50000 — 'D';
50001–200000 — 'C';
200001–1000000 — 'B';
1000001 и выше — 'A'.

Например, кредитополучателю с доходом 25000 нужно назначить категорию 'E', а клиенту, получающему 235000, — 'B'.

Задание 17. Выведите на экран перечень уникальных целей взятия кредита из столбца `purpose`.

Задание 18. Создайте функцию, которая на основании данных из столбца `purpose` сформирует новый столбец `purpose_category`, куда войдут следующие категории:

```
'операции с автомобилем',  
'операции с недвижимостью',  
'проведение свадьбы',  
'получение образования'.
```

Например, если в столбце `purpose` находится подстрока 'на покупку автомобиля', то в столбце `purpose_category` должна появиться строка 'операции с автомобилем'.


Используйте собственную функцию с именем `categorize_purpose()` и метод `apply()`. Изучите данные в столбце `purpose` и определите, какие подстроки помогут вам правильно определить категорию.

Вторая часть

Эта часть работы (шаги 3 и 4) будет проверена вручную ревьюером. Вы можете выполнять любые вычисления и строить визуализации, которые помогут вам ответить на вопросы и сделать выводы.

В шаблоне вы увидите авторское решение первой части проекта (шаги 1 и 2). Сравните его со своим кодом.

Перед тем как приступить к решению второй части проекта, не забудьте выполнить все ячейки с кодом из шагов 1 и 2, чтобы загрузить все нужные данные.

 Если вашу работу отправили на доработку, пожалуйста, не удаляйте в Jupyter-тетрадке комментарии ревьюера. Так ревьюеру будет проще проверить изменения.

Вторая часть

Шаг 3. Исследуйте данные и ответьте на вопросы

Ответы на вопросы можно разместить в ячейках тетрадок Jupyter Notebook с типом `markdown`.

Задание 19. Есть ли зависимость между количеством детей и возвратом кредита в срок?

Задание 20. Есть ли зависимость между семейным положением и возвратом кредита в срок?

Задание 21. Есть ли зависимость между уровнем дохода и возвратом кредита в срок?

Задание 22. Как разные цели кредита влияют на его возврат в срок?

Задание 23. Приведите возможные причины появления пропусков в исходных данных.

Задание 24. Объясните, почему заполнить пропуски медианным значением — лучшее решение для количественных переменных.

Ответы сопроводите интерпретацией — поясните, о чём именно говорит полученный вами результат.

Шаг 4. Напишите общий вывод

Оформление: Задание выполните в Jupyter Notebook. Программный код заполните в ячейках типа `code`, текстовые пояснения — в ячейках типа `markdown`. Примените форматирование и заголовки.