:preprocessing.1

در مرحله اول باید تعداد سطر و ستون دیتاست را به دست آورد.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 268850 entries, 0 to 268849
Data columns (total 49 columns):
 #   Column                    Non-Null Count   Dtype
---  ------                    --------------   -----
 0   regio1                    268850 non-null  object
 1   serviceCharge             261941 non-null  float64
 2   heatingType               223994 non-null  object
 3   telekomTvOffer            236231 non-null  object
 4   telekomHybridUploadSpeed  45020 non-null   float64
 5   newlyConst                268850 non-null  bool
 6   balcony                   268850 non-null  bool
 7   picturecount              268850 non-null  int64
 8   pricetrend                267018 non-null  float64
 9   telekomUploadSpeed        235492 non-null  float64
 10  totalRent                 228333 non-null  float64
 11  yearConstructed           211805 non-null  float64
 12  scoutId                   268850 non-null  int64
 13  noParkSpaces              93052 non-null   float64
 14  firingTypes               211886 non-null  object
 15  hasKitchen                268850 non-null  bool
 16  geo_bln                   268850 non-null  object
 17  cellar                    268850 non-null  bool
 18  yearConstructedRange      211805 non-null  float64
 19  baseRent                  268850 non-null  float64
 20  houseNumber               197832 non-null  object
 21  livingSpace               268850 non-null  float64
 22  geo_krs                   268850 non-null  object
 23  condition                 200361 non-null  object
 24  interiorQual              156185 non-null  object
 25  petsAllowed               154277 non-null  object
 26  street                    268850 non-null  object
 27  streetPlain               197837 non-null  object
 28  lift                      268850 non-null  bool
 29  baseRentRange             268850 non-null  int64
 30  typeOfFlat                232236 non-null  object
 31  geo_plz                   268850 non-null  int64
 32  noRooms                   268850 non-null  float64
 33  thermalChar               162344 non-null  float64
 34  floor                     217541 non-null  float64
 35  numberOfFloors            171118 non-null  float64
```

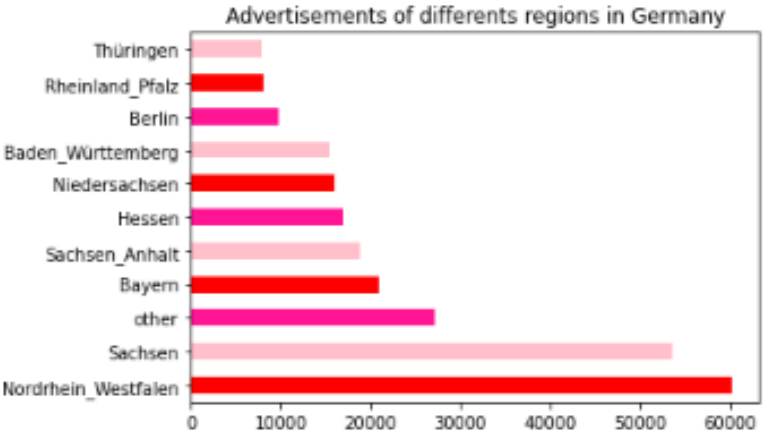
دیتاست شامل 268849 رکورد و 49 فیچر است که تعدادی از انها در بالا نمایش داده شده است.

با حذف ستون‌هایی که بیش از نصف داده های آنها نال است تعداد ستدن ها به 42 کاهش میابد.
مقدار totalrent در برخی رکورد ها برابر 0 است و این مقدار نادرست و رکوردهایی با این شرایط را دراپ می‌کنیم.
ستون هایی که داده های کتگوریکال با تنوع بالا دارند را نیز دراپ می‌کنیم.
علاوه بر آن داده هایی که روی قیمت خانه تاثیر ندارد را هم دراپ میکنیم.
بعد از پاک کردن داده های رکورد های duplicate و outlier برای پر کردن مقادیر نال در ستون های عددی از میانگین و در ستون های کتگوریکال از مود استفاده میکنیم.
برخی از ستون های کتگوریکال دارای تنوع زیادی هستند و تبدیل مستقیم انها به داده باینری ستون های زیادی را به دیتاست اضافه میکند، به همین دلیل موارد پر تکرار را نگه داشته و بقیه را به other تغییر نام می دهیم.

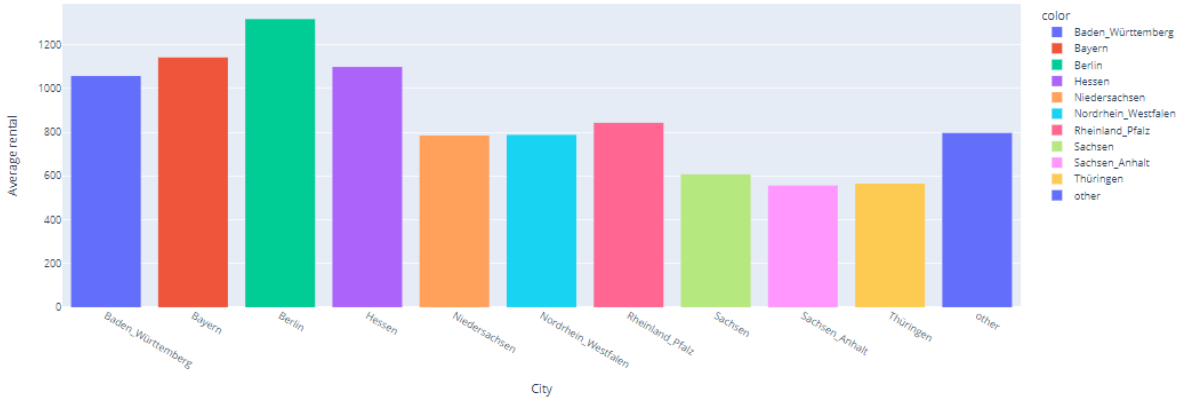
:visualization.2

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 254790 entries, 0 to 268849
Data columns (total 23 columns):
 #   Column              Non-Null Count   Dtype
---  ------              --------------   -----
 0   serviceCharge       254790 non-null  float64
 1   newlyConst          254790 non-null  bool
 2   balcony             254790 non-null  bool
 3   totalRent           254790 non-null  float64
 4   yearConstructed     254790 non-null  float64
 5   hasKitchen          254790 non-null  bool
 6   cellar              254790 non-null  bool
 7   baseRent            254790 non-null  float64
 8   livingSpace         254790 non-null  float64
 9   interiorQual        254790 non-null  object
 10  petsAllowed         254790 non-null  object
 11  lift                254790 non-null  bool
 12  noRooms             254790 non-null  float64
 13  thermalChar         254790 non-null  float64
 14  floor               254790 non-null  float64
 15  numberOfFloors      254790 non-null  float64
 16  garden              254790 non-null  bool
 17  edited_regio1       254790 non-null  object
 18  edited_regio2       254790 non-null  object
 19  edited_regio3       254790 non-null  object
 20  edited_heatingType  254790 non-null  object
 21  edited_condition    254790 non-null  object
 22  edited_typeOfFlat   254790 non-null  object
dtypes: bool(6), float64(9), object(8)
memory usage: 36.4+ MB
```

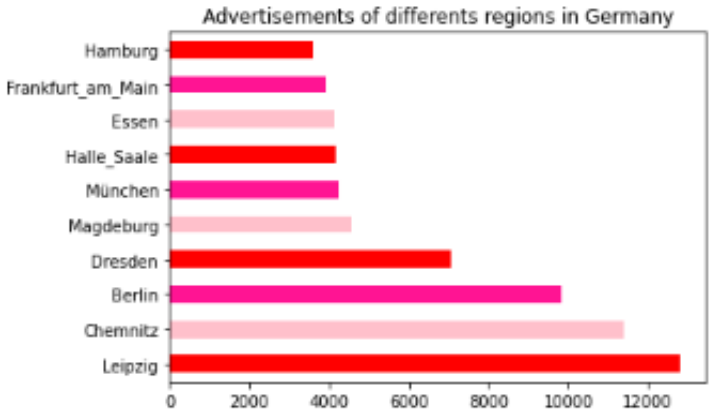
دیتا پس از preprocessing شامل 254790 رکورد و 23 ستون است.



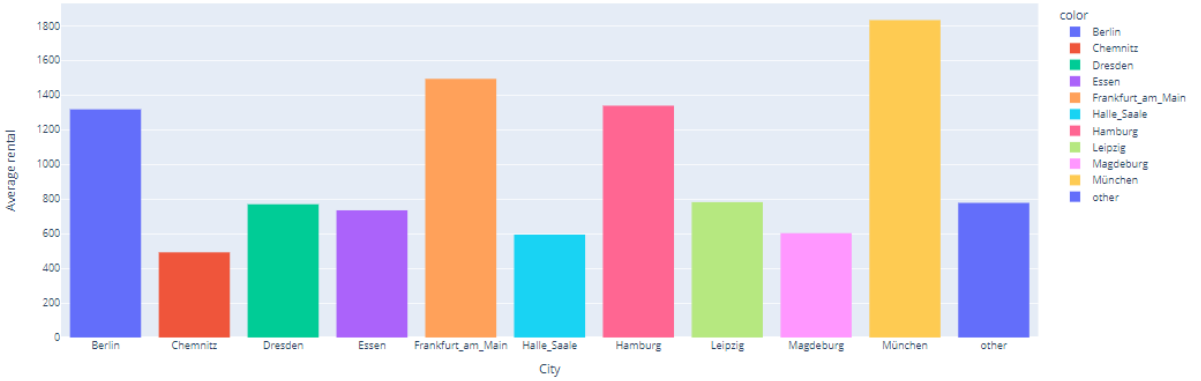
نمودار بالا فراوانی آگهی در شهرهای مختلف از regio1 را نشان می‌دهد



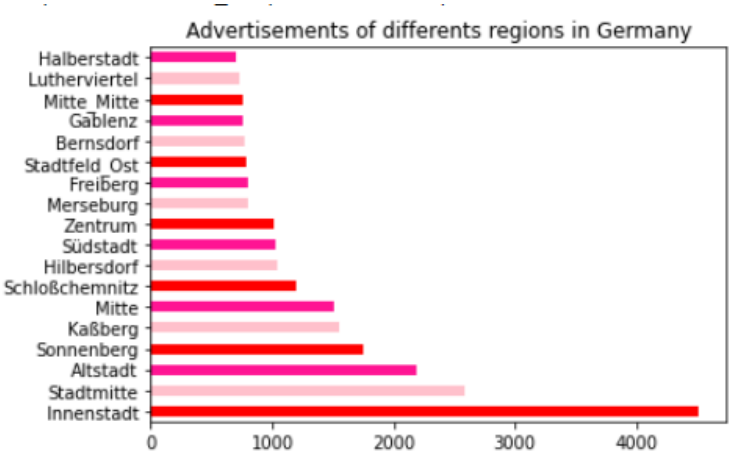
نمودار بالا میانگین قیمت در شهرهای مختلف از regio1 را نشان می‌دهد



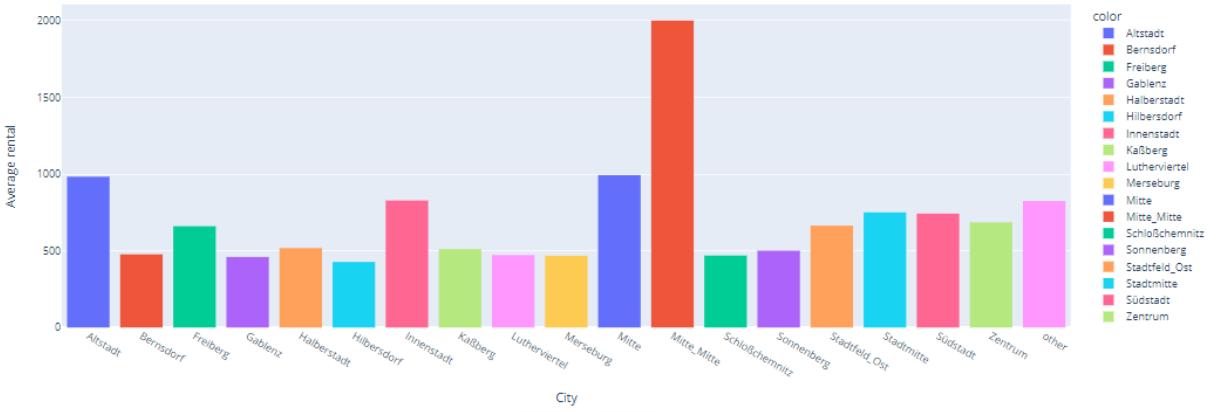
نمودار بالا فراوانی آگهی در شهرهای مختلف از regio2 را نشان می‌دهد.



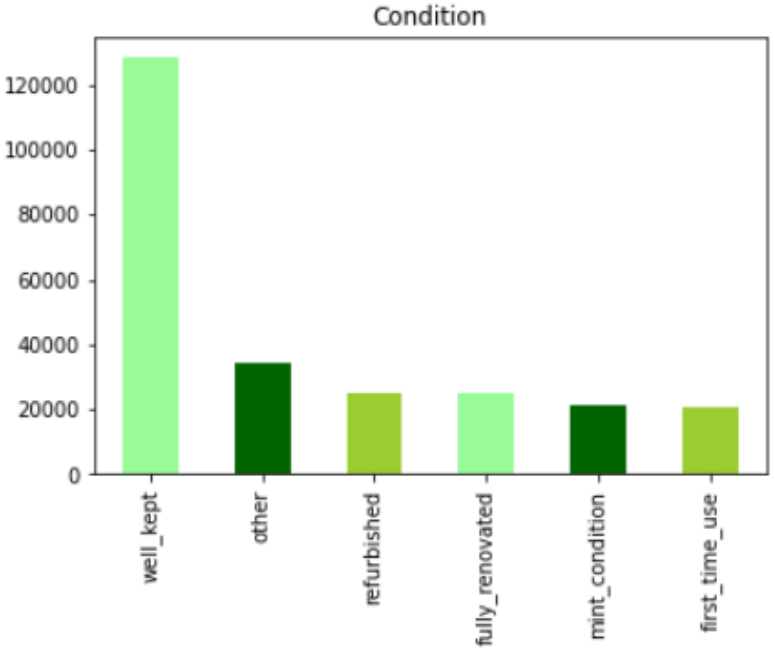
نمودار بالا میانگین قیمت در شهرهای مختلف از regio2 را نشان می‌دهد



نمودار بالا فراوانی آگهی در شهرهای مختلف از regio3 را نشان می‌دهد.



نمودار بالا میانگین قیمت در شهرهای مختلف از regio3 را نشان می‌دهد



نمودار بالا فراوانی وضعیت خانه ها را نشان می‌دهد.

:model.3

در این بخش بعد از تبدیل داده های کتگوریکال به باینری، ستون totalrent را به عنوان تارگت از دیتاست حذف میکنیم. سپس با متون minmax مرحله اسکیل کردن داده هارا انجام می‌دهیم. و linear regression را با دیتای جدید فیت می‌کنیم.

:5 & 4

تابع fillna را به عنوان بخشی از preprocessing ، برای مقایسه ی سه حالت انتخاب کردم.

```
single processing:  0.02665996551513672
multiprocessing:  0.16583251953125
processing with Dask:  0.020489215850830078
```

نتیجه به صورت بالا بود.
در multiprocessing به دلیل اینکه تقسیم دیتا و یکپارچه کردن آن پس از محاسبات زمان گیر است، در این مورد به جای کاهش تایم شاهد افزایش ان هستیم. ولی در dask تایم کاهش یافته است. دلیل عدم استفاده از pyspark سخت افزار ضعیف بود.