

```

{
  "cells": [
    {
      "cell_type": "code",
      "execution_count": 52,
      "id": "84664f05-01b9-464b-9a11-1683e9a9ad84",
      "metadata": {},
      "outputs": [
        {
          "data": {
            "text/html": [
              "<div>\n",
              "<style scoped>\n",
              "  .dataframe tbody tr th:only-of-type {\n",
              "    vertical-align: middle;\n",
              "  }\n",
              "\n",
              "  .dataframe tbody tr th {\n",
              "    vertical-align: top;\n",
              "  }\n",
              "\n",
              "  .dataframe thead th {\n",
              "    text-align: right;\n",
              "  }\n",
              "</style>\n",
              "<table border='1' class='dataframe'>\n",
              "  <thead>\n",
              "    <tr style='text-align: right;'>\n",
              "      <th></th>\n",
              "      <th>id</th>\n",
              "      <th>name</th>\n",
              "      <th>host_id</th>\n",
              "      <th>host_name</th>\n",
              "      <th>neighbourhood_group</th>\n",
              "      <th>neighbourhood</th>\n",
              "      <th>latitude</th>\n",
              "      <th>longitude</th>\n",
              "      <th>room_type</th>\n",
              "      <th>price</th>\n",
              "      <th>minimum_nights</th>\n",
              "      <th>number_of_reviews</th>\n",
              "      <th>last_review</th>\n",
              "      <th>reviews_per_month</th>\n",
              "      <th>calculated_host_listings_count</th>\n",
              "      <th>availability_365</th>\n",
              "    </tr>\n",
              "  </thead>\n",
              "  <tbody>\n",
              "    <tr>\n",
              "      <th>0</th>\n",
              "      <td>2539</td>\n",
              "      <td>Clean & quiet apt home by the park</td>\n",
              "      <td>2787</td>\n",
              "      <td>John</td>\n",
              "      <td>Brooklyn</td>\n",
              "      <td>Kensington</td>\n",
              "      <td>40.64749</td>\n",
              "      <td>-73.97237</td>\n",
              "      <td>Private room</td>\n",
              "      <td>149</td>\n",
              "      <td>1</td>\n",
              "      <td>9</td>\n",
              "      <td>2018-10-19</td>\n",
              "      <td>0.21</td>\n",
              "      <td>6</td>\n",
              "      <td>365</td>\n",
            ]
          }
        ]
      ]
    }
  ]
}

```

```

"      </tr>\n",
"      <tr>\n",
"          <th>1</th>\n",
"          <td>2595</td>\n",
"          <td>Skylit Midtown Castle</td>\n",
"          <td>2845</td>\n",
"          <td>Jennifer</td>\n",
"          <td>Manhattan</td>\n",
"          <td>Midtown</td>\n",
"          <td>40.75362</td>\n",
"          <td>-73.98377</td>\n",
"          <td>Entire home/apt</td>\n",
"          <td>225</td>\n",
"          <td>1</td>\n",
"          <td>45</td>\n",
"          <td>2019-05-21</td>\n",
"          <td>0.38</td>\n",
"          <td>2</td>\n",
"          <td>355</td>\n",
"      </tr>\n",
"      <tr>\n",
"          <th>2</th>\n",
"          <td>3647</td>\n",
"          <td>THE VILLAGE OF HARLEM...NEW YORK !</td>\n",
"          <td>4632</td>\n",
"          <td>Elisabeth</td>\n",
"          <td>Manhattan</td>\n",
"          <td>Harlem</td>\n",
"          <td>40.80902</td>\n",
"          <td>-73.94190</td>\n",
"          <td>Private room</td>\n",
"          <td>150</td>\n",
"          <td>3</td>\n",
"          <td>0</td>\n",
"          <td>NaN</td>\n",
"          <td>NaN</td>\n",
"          <td>1</td>\n",
"          <td>365</td>\n",
"      </tr>\n",
"      <tr>\n",
"          <th>3</th>\n",
"          <td>3831</td>\n",
"          <td>Cozy Entire Floor of Brownstone</td>\n",
"          <td>4869</td>\n",
"          <td>LisaRoxanne</td>\n",
"          <td>Brooklyn</td>\n",
"          <td>Clinton Hill</td>\n",
"          <td>40.68514</td>\n",
"          <td>-73.95976</td>\n",
"          <td>Entire home/apt</td>\n",
"          <td>89</td>\n",
"          <td>1</td>\n",
"          <td>270</td>\n",
"          <td>2019-07-05</td>\n",
"          <td>4.64</td>\n",
"          <td>1</td>\n",
"          <td>194</td>\n",
"      </tr>\n",
"      <tr>\n",
"          <th>4</th>\n",
"          <td>5022</td>\n",
"          <td>Entire Apt: Spacious Studio/Loft by central park</td>\n",
"          <td>7192</td>\n",
"          <td>Laura</td>\n",
"          <td>Manhattan</td>\n",
"          <td>East Harlem</td>\n",
"          <td>40.79851</td>\n",

```

```

"      <td>-73.94399</td>\n",
"      <td>Entire home/apt</td>\n",
"      <td>80</td>\n",
"      <td>10</td>\n",
"      <td>9</td>\n",
"      <td>2018-11-19</td>\n",
"      <td>0.10</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>...</th>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>48890</th>\n",
"      <td>36484665</td>\n",
"      <td>Charming one bedroom - newly renovated rowhouse</td>\n",
"      <td>8232441</td>\n",
"      <td>Sabrina</td>\n",
"      <td>Brooklyn</td>\n",
"      <td>Bedford-Stuyvesant</td>\n",
"      <td>40.67853</td>\n",
"      <td>-73.94995</td>\n",
"      <td>Private room</td>\n",
"      <td>70</td>\n",
"      <td>2</td>\n",
"      <td>0</td>\n",
"      <td>NaN</td>\n",
"      <td>NaN</td>\n",
"      <td>2</td>\n",
"      <td>9</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>48891</th>\n",
"      <td>36485057</td>\n",
"      <td>Affordable room in Bushwick/East Williamsburg</td>\n",
"      <td>6570630</td>\n",
"      <td>Marisol</td>\n",
"      <td>Brooklyn</td>\n",
"      <td>Bushwick</td>\n",
"      <td>40.70184</td>\n",
"      <td>-73.93317</td>\n",
"      <td>Private room</td>\n",
"      <td>40</td>\n",
"      <td>4</td>\n",
"      <td>0</td>\n",
"      <td>NaN</td>\n",
"      <td>NaN</td>\n",
"      <td>2</td>\n",
"      <td>36</td>\n",
"    </tr>\n",

```

```

"      <tr>\n",
"        <th>48892</th>\n",
"        <td>36485431</td>\n",
"        <td>Sunny Studio at Historical Neighborhood</td>\n",
"        <td>23492952</td>\n",
"        <td>Ilgar & Aysel</td>\n",
"        <td>Manhattan</td>\n",
"        <td>Harlem</td>\n",
"        <td>40.81475</td>\n",
"        <td>-73.94867</td>\n",
"        <td>Entire home/apt</td>\n",
"        <td>115</td>\n",
"        <td>10</td>\n",
"        <td>0</td>\n",
"        <td>NaN</td>\n",
"        <td>NaN</td>\n",
"        <td>1</td>\n",
"        <td>27</td>\n",
"      </tr>\n",
"      <tr>\n",
"        <th>48893</th>\n",
"        <td>36485609</td>\n",
"        <td>43rd St. Time Square-cozy single bed</td>\n",
"        <td>30985759</td>\n",
"        <td>Taz</td>\n",
"        <td>Manhattan</td>\n",
"        <td>Hell's Kitchen</td>\n",
"        <td>40.75751</td>\n",
"        <td>-73.99112</td>\n",
"        <td>Shared room</td>\n",
"        <td>55</td>\n",
"        <td>1</td>\n",
"        <td>0</td>\n",
"        <td>NaN</td>\n",
"        <td>NaN</td>\n",
"        <td>6</td>\n",
"        <td>2</td>\n",
"      </tr>\n",
"      <tr>\n",
"        <th>48894</th>\n",
"        <td>36487245</td>\n",
"        <td>Trendy duplex in the very heart of Hell's Kitchen</td>\n",
"        <td>68119814</td>\n",
"        <td>Christophe</td>\n",
"        <td>Manhattan</td>\n",
"        <td>Hell's Kitchen</td>\n",
"        <td>40.76404</td>\n",
"        <td>-73.98933</td>\n",
"        <td>Private room</td>\n",
"        <td>90</td>\n",
"        <td>7</td>\n",
"        <td>0</td>\n",
"        <td>NaN</td>\n",
"        <td>NaN</td>\n",
"        <td>1</td>\n",
"        <td>23</td>\n",
"      </tr>\n",
"    </tbody>\n",
"</table>\n",
"<p>48895 rows x 16 columns</p>\n",
"</div>"

```

```

],
"text/plain": [
  "      id                                     name      host_id
\\n",
  "0      2539                                Clean & quiet apt home by the park      2787
\n",

```

```

"1          2595          Skylit Midtown Castle          2845
\n",
"2          3647          THE VILLAGE OF HARLEM....NEW YORK !          4632
\n",
"3          3831          Cozy Entire Floor of Brownstone          4869
\n",
"4          5022  Entire Apt: Spacious Studio/Loft by central park          7192
\n",
"...          ...          ...          ...
\n",
"48890  36484665  Charming one bedroom - newly renovated rowhouse          8232441
\n",
"48891  36485057  Affordable room in Bushwick/East Williamsburg          6570630
\n",
"48892  36485431          Sunny Studio at Historical Neighborhood          23492952
\n",
"48893  36485609          43rd St. Time Square-cozy single bed          30985759
\n",
"48894  36487245  Trendy duplex in the very heart of Hell's Kitchen          68119814
\n",
"\n",
"          host_name  neighbourhood_group          neighbourhood  latitude  \\\n",
"0          John          Brooklyn          Kensington  40.64749  \n",
"1          Jennifer          Manhattan          Midtown  40.75362  \n",
"2          Elisabeth          Manhattan          Harlem  40.80902  \n",
"3          LisaRoxanne          Brooklyn          Clinton Hill  40.68514  \n",
"4          Laura          Manhattan          East Harlem  40.79851  \n",
"...          ...          ...          ...          \n",
"48890          Sabrina          Brooklyn  Bedford-Stuyvesant  40.67853  \n",
"48891          Marisol          Brooklyn          Bushwick  40.70184  \n",
"48892  Ilgar & Aysel          Manhattan          Harlem  40.81475  \n",
"48893          Taz          Manhattan          Hell's Kitchen  40.75751  \n",
"48894          Christophe          Manhattan          Hell's Kitchen  40.76404  \n",
"\n",
"          longitude          room_type  price  minimum_nights  number_of_reviews
\\\n",
"0          -73.97237  Private room  149          1          9
\n",
"1          -73.98377  Entire home/apt  225          1          45
\n",
"2          -73.94190  Private room  150          3          0
\n",
"3          -73.95976  Entire home/apt  89          1          270
\n",
"4          -73.94399  Entire home/apt  80          10          9
\n",
"...          ...          ...          ...          ...
\n",
"48890  -73.94995  Private room  70          2          0
\n",
"48891  -73.93317  Private room  40          4          0
\n",
"48892  -73.94867  Entire home/apt  115          10          0
\n",
"48893  -73.99112  Shared room  55          1          0
\n",
"48894  -73.98933  Private room  90          7          0
\n",
"\n",
"          last_review  reviews_per_month  calculated_host_listings_count  \\\n",
"0          2018-10-19          0.21          6          \n",
"1          2019-05-21          0.38          2          \n",
"2          NaN          NaN          1          \n",
"3          2019-07-05          4.64          1          \n",
"4          2018-11-19          0.10          1          \n",
"...          ...          ...          \n",
"48890          NaN          NaN          2          \n",

```

```

"48891      NaN      NaN      2  \n",
"48892      NaN      NaN      1  \n",
"48893      NaN      NaN      6  \n",
"48894      NaN      NaN      1  \n",
"\n",
"      availability_365  \n",
"0      365  \n",
"1      355  \n",
"2      365  \n",
"3      194  \n",
"4      0  \n",
"...      ...  \n",
"48890      9  \n",
"48891      36  \n",
"48892      27  \n",
"48893      2  \n",
"48894      23  \n",
"\n",
"[48895 rows x 16 columns]"
]
},
"execution_count": 52,
"metadata": {},
"output_type": "execute_result"
}
],
"source": [
"import pandas as pd\n",
"df=pd.read_csv(\"E:\\chromedownload\\AB_NYC_2019.csv\")\n",
"df"
]
},
{
"cell_type": "code",
"execution_count": 179,
"id": "71bfe927-1b01-4926-809b-695beb69a9c7",
"metadata": {},
"outputs": [
{
"name": "stdout",
"output_type": "stream",
"text": [
"Requirement already satisfied: ydata-profiling in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (4.10.0)\n",
"Requirement already satisfied: scipy<1.14,>=1.4.1 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from ydata-profiling) (1.11.4)\n",
"Requirement already satisfied: pandas!=1.4.0,<3,>1.1 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from ydata-profiling) (2.1.4)\n",
"Requirement already satisfied: matplotlib<3.10,>=3.5 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from ydata-profiling) (3.8.0)\n",
"Requirement already satisfied: pydantic>=2 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from ydata-profiling) (2.9.1)\n",
"Requirement already satisfied: PyYAML<6.1,>=5.0.0 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from ydata-profiling) (6.0.1)\n",
"Requirement already satisfied: jinja2<3.2,>=2.11.1 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from ydata-profiling) (3.1.3)\n",
"Requirement already satisfied: visions<0.7.7,>=0.7.5 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from visions[type_image_path]
<0.7.7,>=0.7.5->ydata-profiling) (0.7.6)\n",
"Requirement already satisfied: numpy<2.2,>=1.16.0 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from ydata-profiling) (1.26.4)\n",
"Requirement already satisfied: htmlmin==0.1.12 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from ydata-profiling) (0.1.12)\n",
"Requirement already satisfied: phik<0.13,>=0.11.1 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from ydata-profiling) (0.12.4)\n",
"Requirement already satisfied: requests<3,>=2.24.0 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from ydata-profiling) (2.31.0)\n",

```

```

"Requirement already satisfied: tqdm<5,>=4.48.2 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from ydata-profiling) (4.65.0)\n",
"Requirement already satisfied: seaborn<0.14,>=0.10.1 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from ydata-profiling) (0.12.2)\n",
"Requirement already satisfied: multimethod<2,>=1.4 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from ydata-profiling) (1.12)\n",
"Requirement already satisfied: statsmodels<1,>=0.13.2 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from ydata-profiling) (0.14.0)\n",
"Requirement already satisfied: typeguard<5,>=3 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from ydata-profiling) (4.3.0)\n",
"Requirement already satisfied: imagehash==4.3.1 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from ydata-profiling) (4.3.1)\n",
"Requirement already satisfied: wordcloud>=1.9.3 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from ydata-profiling) (1.9.3)\n",
"Requirement already satisfied: dacite>=1.8 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from ydata-profiling) (1.8.1)\n",
"Requirement already satisfied: numba<1,>=0.56.0 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from ydata-profiling) (0.59.0)\n",
"Requirement already satisfied: PyWavelets in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from imagehash==4.3.1->ydata-profiling)
(1.5.0)\n",
"Requirement already satisfied: pillow in c:\\users\\sharm\\anaconda3\\lib\\site-
packages (from imagehash==4.3.1->ydata-profiling) (10.2.0)\n",
"Requirement already satisfied: MarkupSafe>=2.0 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from jinja2<3.2,>=2.11.1->ydata-
profiling) (2.1.3)\n",
"Requirement already satisfied: contourpy>=1.0.1 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from matplotlib<3.10,>=3.5->ydata-
profiling) (1.2.0)\n",
"Requirement already satisfied: cycler>=0.10 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from matplotlib<3.10,>=3.5->ydata-
profiling) (0.11.0)\n",
"Requirement already satisfied: fonttools>=4.22.0 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from matplotlib<3.10,>=3.5->ydata-
profiling) (4.25.0)\n",
"Requirement already satisfied: kiwisolver>=1.0.1 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from matplotlib<3.10,>=3.5->ydata-
profiling) (1.4.4)\n",
"Requirement already satisfied: packaging>=20.0 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from matplotlib<3.10,>=3.5->ydata-
profiling) (23.1)\n",
"Requirement already satisfied: pyparsing>=2.3.1 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from matplotlib<3.10,>=3.5->ydata-
profiling) (3.0.9)\n",
"Requirement already satisfied: python-dateutil>=2.7 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from matplotlib<3.10,>=3.5->ydata-
profiling) (2.8.2)\n",
"Requirement already satisfied: llvmlite<0.43,>=0.42.0dev0 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from numba<1,>=0.56.0->ydata-profiling)
(0.42.0)\n",
"Requirement already satisfied: pytz>=2020.1 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from pandas!=1.4.0,<3,>1.1->ydata-
profiling) (2023.3.post1)\n",
"Requirement already satisfied: tzdata>=2022.1 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from pandas!=1.4.0,<3,>1.1->ydata-
profiling) (2023.3)\n",
"Requirement already satisfied: joblib>=0.14.1 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from phik<0.13,>=0.11.1->ydata-
profiling) (1.1.1)\n",
"Requirement already satisfied: annotated-types>=0.6.0 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from pydantic>=2->ydata-profiling)
(0.7.0)\n",
"Requirement already satisfied: pydantic-core==2.23.3 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from pydantic>=2->ydata-profiling)
(2.23.3)\n",
"Requirement already satisfied: typing-extensions>=4.6.1 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from pydantic>=2->ydata-profiling)

```

```

(4.12.2)\n",
  "Requirement already satisfied: charset-normalizer<4,>=2 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from requests<3,>=2.24.0->ydata-
profiling) (2.0.4)\n",
  "Requirement already satisfied: idna<4,>=2.5 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from requests<3,>=2.24.0->ydata-
profiling) (3.4)\n",
  "Requirement already satisfied: urllib3<3,>=1.21.1 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from requests<3,>=2.24.0->ydata-
profiling) (1.26.20)\n",
  "Requirement already satisfied: certifi>=2017.4.17 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from requests<3,>=2.24.0->ydata-
profiling) (2024.6.2)\n",
  "Requirement already satisfied: patsy>=0.5.2 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from statsmodels<1,>=0.13.2->ydata-
profiling) (0.5.3)\n",
  "Requirement already satisfied: colorama in c:\\users\\sharm\\anaconda3\\lib\\site-
packages (from tqdm<5,>=4.48.2->ydata-profiling) (0.4.6)\n",
  "Requirement already satisfied: attrs>=19.3.0 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from visions<0.7.7,>=0.7.5-
>visions[type_image_path]<0.7.7,>=0.7.5->ydata-profiling) (23.1.0)\n",
  "Requirement already satisfied: networkx>=2.4 in
c:\\users\\sharm\\anaconda3\\lib\\site-packages (from visions<0.7.7,>=0.7.5-
>visions[type_image_path]<0.7.7,>=0.7.5->ydata-profiling) (3.1)\n",
  "Requirement already satisfied: six in c:\\users\\sharm\\anaconda3\\lib\\site-
packages (from patsy>=0.5.2->statsmodels<1,>=0.13.2->ydata-profiling) (1.16.0)\n",
  "Note: you may need to restart the kernel to use updated packages.\n"
]
},
],
"source": [
  "pip install ydata-profiling"
],
},
{
  "cell_type": "code",
  "execution_count": 192,
  "id": "3c43db55-175b-45d9-8aff-604c191c5f79",
  "metadata": {},
  "outputs": [
    {
      "data": {
        "application/vnd.jupyter.widget-view+json": {
          "model_id": "6151c7d5502c4121a5d741bd5a716569",
          "version_major": 2,
          "version_minor": 0
        },
        "text/plain": [
          "Summarize dataset:   0%|          | 0/5 [00:00<?, ?it/s]"
        ]
      },
      "metadata": {},
      "output_type": "display_data"
    },
    {
      "data": {
        "application/vnd.jupyter.widget-view+json": {
          "model_id": "2f9345511a3a47eeb0b1c5b485da90c9",
          "version_major": 2,
          "version_minor": 0
        },
        "text/plain": [
          "Generate report structure:   0%|          | 0/1 [00:00<?, ?it/s]"
        ]
      },
      "metadata": {},
      "output_type": "display_data"
    }
  ]
}

```



```

},
{
  "data": {
    "application/vnd.jupyter.widget-view+json": {
      "model_id": "55e5c6b344cb4cc2a04eb7f62d26c6f3",
      "version_major": 2,
      "version_minor": 0
    },
    "text/plain": [
      "Render HTML: 0%|          | 0/1 [00:00<?, ?it/s]"
    ]
  },
  "metadata": {},
  "output_type": "display_data"
},
{
  "data": {
    "application/vnd.jupyter.widget-view+json": {
      "model_id": "0fe9ef12c59c4f15a2bc6ff948cb9737",
      "version_major": 2,
      "version_minor": 0
    },
    "text/plain": [
      "Export report to file: 0%|          | 0/1 [00:00<?, ?it/s]"
    ]
  },
  "metadata": {},
  "output_type": "display_data"
}
],
"source": [
  "from ydata_profiling import ProfileReport\n",
  "profile = ProfileReport(df, title=\"AB_NYC Profiling Report\", explorative=True)\n",
  "profile.to_file(\"AB_NYC_report.html\")"
]
},
{
  "cell_type": "code",
  "execution_count": 191,
  "id": "fa0387a0-5ee3-4fdc-bccb-311599b372ea",
  "metadata": {},
  "outputs": [
    {
      "data": {
        "text/plain": [
          "id                                0\n",
          "name                             0\n",
          "host_id                           0\n",
          "host_name                         0\n",
          "neighbourhood_group              0\n",
          "neighbourhood                    0\n",
          "latitude                         0\n",
          "longitude                        0\n",
          "room_type                        0\n",
          "price                            0\n",
          "minimum_nights                   0\n",
          "number_of_reviews                 0\n",
          "last_review                      10037\n",
          "reviews_per_month                 0\n",
          "calculated_host_listings_count    0\n",
          "availability_365                  0\n",
          "dtype: int64"
        ]
      },
      "execution_count": 191,
      "metadata": {},
      "output_type": "execute_result"
    }
  ]
}

```

```

    }
  ],
  "source": [
    "df.isnull().sum()"
  ]
},
{
  "cell_type": "markdown",
  "id": "6d07d00f-872d-4324-8c3b-13c78ed4ee54",
  "metadata": {},
  "source": [
    "### Handling missing values"
  ]
},
{
  "cell_type": "code",
  "execution_count": 201,
  "id": "66ecea04-543b-41be-992d-ec665286a057",
  "metadata": {},
  "outputs": [],
  "source": [
    "df[\"last_review\"] = pd.to_datetime(df[\"last_review\"], dayfirst=True,
errors='coerce')"
  ]
},
{
  "cell_type": "code",
  "execution_count": 207,
  "id": "5b7090e6-bef2-4bb4-990f-12f67ed1d34e",
  "metadata": {},
  "outputs": [],
  "source": [
    "df['last_review'] = df['last_review'].ffill()\n",
    "df.ffill(inplace=True)"
  ]
},
{
  "cell_type": "code",
  "execution_count": 209,
  "id": "3d2fcb99-af97-4382-a68c-b2e2733cb6fd",
  "metadata": {},
  "outputs": [],
  "source": [
    "df.dropna(subset=[\"host_name\"], inplace=True)"
  ]
},
{
  "cell_type": "code",
  "execution_count": 211,
  "id": "e00db1a3-1278-4dad-bd8b-fcd3b5892e53",
  "metadata": {},
  "outputs": [],
  "source": [
    "df['reviews_per_month'].fillna(0, inplace=True)"
  ]
},
{
  "cell_type": "code",
  "execution_count": 213,
  "id": "0fe4f4a5-502a-4e1e-9be4-d8e643dcd387",
  "metadata": {},
  "outputs": [],
  "source": [
    "df.dropna(subset=[\"name\"], inplace=True)"
  ]
},
}

```

```

"cell_type": "code",
"execution_count": 215,
"id": "8a29f47c-a303-4c9f-b319-1ad58d353b1b",
"metadata": {},
"outputs": [
  {
    "data": {
      "text/plain": [
        "id                0\n",
        "name              0\n",
        "host_id            0\n",
        "host_name          0\n",
        "neighbourhood_group 0\n",
        "neighbourhood      0\n",
        "latitude           0\n",
        "longitude           0\n",
        "room_type          0\n",
        "price              0\n",
        "minimum_nights     0\n",
        "number_of_reviews   0\n",
        "last_review         0\n",
        "reviews_per_month   0\n",
        "calculated_host_listings_count 0\n",
        "availability_365    0\n",
        "dtype: int64"
      ]
    },
    "execution_count": 215,
    "metadata": {},
    "output_type": "execute_result"
  },
  {
    "source": [
      "df.isnull().sum()"
    ]
  },
  {
    "cell_type": "code",
    "execution_count": 217,
    "id": "b76d81ca-d031-4f2c-b0ff-4467e60d4275",
    "metadata": {},
    "outputs": [
      {
        "data": {
          "text/plain": [
            "0"
          ]
        },
        "execution_count": 217,
        "metadata": {},
        "output_type": "execute_result"
      }
    ],
    "source": [
      "df.duplicated().sum()"
    ]
  },
  {
    "cell_type": "code",
    "execution_count": 219,
    "id": "40d0ff9f-5e61-49b8-931b-c11d85543150",
    "metadata": {},
    "outputs": [
      {
        "data": {
          "text/plain": [
            "id                float64\n",

```

```

        "name"                object\n",
        "host_id"             float64\n",
        "host_name"           object\n",
        "neighbourhood_group" object\n",
        "neighbourhood"       object\n",
        "latitude"            float64\n",
        "longitude"           float64\n",
        "room_type"           object\n",
        "price"               float64\n",
        "minimum_nights"      float64\n",
        "number_of_reviews"    float64\n",
        "last_review"         datetime64[ns]\n",
        "reviews_per_month"    float64\n",
        "calculated_host_listings_count" float64\n",
        "availability_365"     float64\n",
        "dtype: object"
    ]
},
"execution_count": 219,
"metadata": {},
"output_type": "execute_result"
}
],
"source": [
    "df.dtypes"
]
},
{
    "cell_type": "code",
    "execution_count": 242,
    "id": "3ca0f285-25a1-4313-9c34-72573e310a87",
    "metadata": {},
    "outputs": [
        {
            "data": {
                "text/html": [
                    "<div>\n",
                    "<style scoped>\n",
                    "    .dataframe tbody tr th:only-of-type {\n",
                    "        vertical-align: middle;\n",
                    "    }\n",
                    "\n",
                    "    .dataframe tbody tr th {\n",
                    "        vertical-align: top;\n",
                    "    }\n",
                    "\n",
                    "    .dataframe thead th {\n",
                    "        text-align: right;\n",
                    "    }\n",
                    "</style>\n",
                    "<table border='1' class='dataframe'>\n",
                    "  <thead>\n",
                    "    <tr style='text-align: right;'>\n",
                    "      <th></th>\n",
                    "      <th>id</th>\n",
                    "      <th>name</th>\n",
                    "      <th>host_id</th>\n",
                    "      <th>host_name</th>\n",
                    "      <th>neighbourhood_group</th>\n",
                    "      <th>neighbourhood</th>\n",
                    "      <th>latitude</th>\n",
                    "      <th>longitude</th>\n",
                    "      <th>room_type</th>\n",
                    "      <th>price</th>\n",
                    "      <th>minimum_nights</th>\n",
                    "      <th>number_of_reviews</th>\n",
                    "      <th>last_review</th>\n",

```

```

"      <th>reviews_per_month</th>\n",
"      <th>calculated_host_listings_count</th>\n",
"      <th>availability_365</th>\n",
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"    <tr>\n",
"      <th>0</th>\n",
"      <td>-1.731876</td>\n",
"      <td>Clean & quiet apt home by the park</td>\n",
"      <td>-0.860166</td>\n",
"      <td>John</td>\n",
"      <td>brooklyn</td>\n",
"      <td>Kensington</td>\n",
"      <td>-1.493773</td>\n",
"      <td>-0.437612</td>\n",
"      <td>private room</td>\n",
"      <td>-0.015570</td>\n",
"      <td>-0.300329</td>\n",
"      <td>-0.320388</td>\n",
"      <td>2018-10-19</td>\n",
"      <td>-0.551649</td>\n",
"      <td>-0.034837</td>\n",
"      <td>1.916263</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>1</th>\n",
"      <td>-1.731871</td>\n",
"      <td>Skylit Midtown Castle</td>\n",
"      <td>-0.860165</td>\n",
"      <td>Jennifer</td>\n",
"      <td>manhattan</td>\n",
"      <td>Midtown</td>\n",
"      <td>0.452597</td>\n",
"      <td>-0.684585</td>\n",
"      <td>entire home/apt</td>\n",
"      <td>0.300794</td>\n",
"      <td>-0.300329</td>\n",
"      <td>0.487703</td>\n",
"      <td>2019-05-21</td>\n",
"      <td>-0.445217</td>\n",
"      <td>-0.156180</td>\n",
"      <td>1.840280</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>2</th>\n",
"      <td>-1.731775</td>\n",
"      <td>THE VILLAGE OF HARLEM....NEW YORK !</td>\n",
"      <td>-0.860142</td>\n",
"      <td>Elisabeth</td>\n",
"      <td>manhattan</td>\n",
"      <td>Harlem</td>\n",
"      <td>1.468605</td>\n",
"      <td>0.222498</td>\n",
"      <td>private room</td>\n",
"      <td>-0.011407</td>\n",
"      <td>-0.200426</td>\n",
"      <td>-0.522411</td>\n",
"      <td>2019-05-21</td>\n",
"      <td>-0.683125</td>\n",
"      <td>-0.186516</td>\n",
"      <td>1.916263</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>3</th>\n",
"      <td>-1.731758</td>\n",
"      <td>Cozy Entire Floor of Brownstone</td>\n",

```

```

"      <td>-0.860139</td>\n",
"      <td>LisaRoxanne</td>\n",
"      <td>brooklyn</td>\n",
"      <td>Clinton Hill</td>\n",
"      <td>-0.803291</td>\n",
"      <td>-0.164426</td>\n",
"      <td>entire home/apt</td>\n",
"      <td>-0.265330</td>\n",
"      <td>-0.300329</td>\n",
"      <td>5.538271</td>\n",
"      <td>2019-07-05</td>\n",
"      <td>2.221862</td>\n",
"      <td>-0.186516</td>\n",
"      <td>0.616965</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>4</th>\n",
"      <td>-1.731650</td>\n",
"      <td>Entire Apt: Spacious Studio/Loft by central park</td>\n",
"      <td>-0.860110</td>\n",
"      <td>Laura</td>\n",
"      <td>manhattan</td>\n",
"      <td>East Harlem</td>\n",
"      <td>1.275857</td>\n",
"      <td>0.177220</td>\n",
"      <td>entire home/apt</td>\n",
"      <td>-0.302795</td>\n",
"      <td>0.149232</td>\n",
"      <td>-0.320388</td>\n",
"      <td>2018-11-19</td>\n",
"      <td>-0.620518</td>\n",
"      <td>-0.186516</td>\n",
"      <td>-0.857091</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>...</th>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>48890</th>\n",
"      <td>1.589881</td>\n",
"      <td>Charming one bedroom - newly renovated rowhouse</td>\n",
"      <td>-0.755494</td>\n",
"      <td>Sabrina</td>\n",
"      <td>brooklyn</td>\n",
"      <td>Bedford-Stuyvesant</td>\n",
"      <td>-0.924515</td>\n",
"      <td>0.048101</td>\n",
"      <td>private room</td>\n",
"      <td>-0.344421</td>\n",
"      <td>-0.250377</td>\n",
"      <td>-0.522411</td>\n",

```

```

"      <td>2019-07-08</td>\n",
"      <td>-0.683125</td>\n",
"      <td>-0.156180</td>\n",
"      <td>-0.788707</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>48891</th>\n",
"      <td>1.589917</td>\n",
"      <td>Affordable room in Bushwick/East Williamsburg</td>\n",
"      <td>-0.776630</td>\n",
"      <td>Marisol</td>\n",
"      <td>brooklyn</td>\n",
"      <td>Bushwick</td>\n",
"      <td>-0.497022</td>\n",
"      <td>0.411628</td>\n",
"      <td>private room</td>\n",
"      <td>-0.469302</td>\n",
"      <td>-0.150475</td>\n",
"      <td>-0.522411</td>\n",
"      <td>2019-07-08</td>\n",
"      <td>-0.683125</td>\n",
"      <td>-0.156180</td>\n",
"      <td>-0.583555</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>48892</th>\n",
"      <td>1.589951</td>\n",
"      <td>Sunny Studio at Historical Neighborhood</td>\n",
"      <td>-0.561397</td>\n",
"      <td>Ilgar & Aysel</td>\n",
"      <td>manhattan</td>\n",
"      <td>Harlem</td>\n",
"      <td>1.573690</td>\n",
"      <td>0.075831</td>\n",
"      <td>entire home/apt</td>\n",
"      <td>-0.157101</td>\n",
"      <td>0.149232</td>\n",
"      <td>-0.522411</td>\n",
"      <td>2019-07-08</td>\n",
"      <td>-0.683125</td>\n",
"      <td>-0.186516</td>\n",
"      <td>-0.651939</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>48893</th>\n",
"      <td>1.589967</td>\n",
"      <td>43rd St. Time Square-cozy single bed</td>\n",
"      <td>-0.466096</td>\n",
"      <td>Taz</td>\n",
"      <td>manhattan</td>\n",
"      <td>Hell's Kitchen</td>\n",
"      <td>0.523938</td>\n",
"      <td>-0.843818</td>\n",
"      <td>shared room</td>\n",
"      <td>-0.406862</td>\n",
"      <td>-0.300329</td>\n",
"      <td>-0.522411</td>\n",
"      <td>2019-07-08</td>\n",
"      <td>-0.683125</td>\n",
"      <td>-0.034837</td>\n",
"      <td>-0.841895</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>48894</th>\n",
"      <td>1.590116</td>\n",
"      <td>Trendy duplex in the very heart of Hell's Kitchen</td>\n",
"      <td>0.006208</td>\n",

```

```

"      <td>Christophe</td>\n",
"      <td>manhattan</td>\n",
"      <td>Hell's Kitchen</td>\n",
"      <td>0.643695</td>\n",
"      <td>-0.805039</td>\n",
"      <td>private room</td>\n",
"      <td>-0.261168</td>\n",
"      <td>-0.000622</td>\n",
"      <td>-0.522411</td>\n",
"      <td>2019-07-08</td>\n",
"      <td>-0.683125</td>\n",
"      <td>-0.186516</td>\n",
"      <td>-0.682332</td>\n",
"    </tr>\n",
"  </tbody>\n",
"</table>\n",
"<p>48858 rows x 16 columns</p>\n",
"</div>"

```

```
],
```

```
"text/plain": [
```

```

"      id                                     name      host_id
\\n",
"0      -1.731876                             Clean & quiet apt home by the park -0.860166
\n",
"1      -1.731871                             Skylit Midtown Castle -0.860165
\n",
"2      -1.731775                             THE VILLAGE OF HARLEM....NEW YORK ! -0.860142
\n",
"3      -1.731758                             Cozy Entire Floor of Brownstone -0.860139
\n",
"4      -1.731650      Entire Apt: Spacious Studio/Loft by central park -0.860110
\n",
"...      ...                                     ...      ...
\n",
"48890  1.589881      Charming one bedroom - newly renovated rowhouse -0.755494
\n",
"48891  1.589917      Affordable room in Bushwick/East Williamsburg -0.776630
\n",
"48892  1.589951      Sunny Studio at Historical Neighborhood -0.561397
\n",
"48893  1.589967      43rd St. Time Square-cozy single bed -0.466096
\n",
"48894  1.590116      Trendy duplex in the very heart of Hell's Kitchen  0.006208
\n",

```

```

"\n",
"      host_name neighbourhood_group      neighbourhood      latitude  \\n",
"0      John      brooklyn      Kensington -1.493773  \n",
"1      Jennifer      manhattan      Midtown  0.452597  \n",
"2      Elisabeth      manhattan      Harlem  1.468605  \n",
"3      LisaRoxanne      brooklyn      Clinton Hill -0.803291  \n",
"4      Laura      manhattan      East Harlem  1.275857  \n",
"...      ...      ...      ...      \n",
"48890      Sabrina      brooklyn      Bedford-Stuyvesant -0.924515  \n",
"48891      Marisol      brooklyn      Bushwick -0.497022  \n",
"48892      Ilgar & Aysel      manhattan      Harlem  1.573690  \n",
"48893      Taz      manhattan      Hell's Kitchen  0.523938  \n",
"48894      Christophe      manhattan      Hell's Kitchen  0.643695  \n",

```

```

"\n",
"      longitude      room_type      price      minimum_nights  \\n",
"0      -0.437612      private room -0.015570      -0.300329  \n",
"1      -0.684585      entire home/apt 0.300794      -0.300329  \n",
"2      0.222498      private room -0.011407      -0.200426  \n",
"3      -0.164426      entire home/apt -0.265330      -0.300329  \n",
"4      0.177220      entire home/apt -0.302795      0.149232  \n",
"...      ...      ...      ...      \n",
"48890  0.048101      private room -0.344421      -0.250377  \n",
"48891  0.411628      private room -0.469302      -0.150475  \n",

```



```

"48892    0.075831    entire home/apt -0.157101    0.149232    \n",
"48893    -0.843818    shared room -0.406862    -0.300329    \n",
"48894    -0.805039    private room -0.261168    -0.000622    \n",
"\n",
"    number_of_reviews    last_review    reviews_per_month    \\\n",
"0    -0.320388    2018-10-19    -0.551649    \n",
"1    0.487703    2019-05-21    -0.445217    \n",
"2    -0.522411    2019-05-21    -0.683125    \n",
"3    5.538271    2019-07-05    2.221862    \n",
"4    -0.320388    2018-11-19    -0.620518    \n",
"    ...    ...    ...    \n",
"48890    -0.522411    2019-07-08    -0.683125    \n",
"48891    -0.522411    2019-07-08    -0.683125    \n",
"48892    -0.522411    2019-07-08    -0.683125    \n",
"48893    -0.522411    2019-07-08    -0.683125    \n",
"48894    -0.522411    2019-07-08    -0.683125    \n",
"\n",
"    calculated_host_listings_count    availability_365    \n",
"0    -0.034837    1.916263    \n",
"1    -0.156180    1.840280    \n",
"2    -0.186516    1.916263    \n",
"3    -0.186516    0.616965    \n",
"4    -0.186516    -0.857091    \n",
"    ...    ...    \n",
"48890    -0.156180    -0.788707    \n",
"48891    -0.156180    -0.583555    \n",
"48892    -0.186516    -0.651939    \n",
"48893    -0.034837    -0.841895    \n",
"48894    -0.186516    -0.682332    \n",
"\n",
"[48858 rows x 16 columns]"
]
},
"execution_count": 242,
"metadata": {},
"output_type": "execute_result"
}
],
"source": [
"df"
]
},
{
"cell_type": "code",
"execution_count": 221,
"id": "30af2228-a490-43fa-9e20-6452d4b3c404",
"metadata": {},
"outputs": [
{
"data": {
"text/html": [
"<div>\n",
"<style scoped>\n",
"    .dataframe tbody tr th:only-of-type {\n",
"        vertical-align: middle;\n",
"    }\n",
"\n",
"    .dataframe tbody tr th {\n",
"        vertical-align: top;\n",
"    }\n",
"\n",
"    .dataframe thead th {\n",
"        text-align: right;\n",
"    }\n",
"</style>\n",
"<table border='1' class='dataframe'>\n",
"  <thead>\n",

```

```

"      <tr style="text-align: right;">\n",
"        <th></th>\n",
"        <th>id</th>\n",
"        <th>host_id</th>\n",
"        <th>latitude</th>\n",
"        <th>longitude</th>\n",
"        <th>price</th>\n",
"        <th>minimum_nights</th>\n",
"        <th>number_of_reviews</th>\n",
"        <th>last_review</th>\n",
"        <th>reviews_per_month</th>\n",
"        <th>calculated_host_listings_count</th>\n",
"        <th>availability_365</th>\n",
"      </tr>\n",
"    </thead>\n",
"    <tbody>\n",
"      <tr>\n",
"        <th>count</th>\n",
"        <td>4.885800e+04</td>\n",
"        <td>4.885800e+04</td>\n",
"        <td>4.885800e+04</td>\n",
"        <td>4.885800e+04</td>\n",
"        <td>4.885800e+04</td>\n",
"        <td>4.885800e+04</td>\n",
"        <td>4.885800e+04</td>\n",
"        <td>48858</td>\n",
"        <td>4.885800e+04</td>\n",
"        <td>4.885800e+04</td>\n",
"        <td>4.885800e+04</td>\n",
"      </tr>\n",
"      <tr>\n",
"        <th>mean</th>\n",
"        <td>-1.489205e-16</td>\n",
"        <td>-3.723012e-17</td>\n",
"        <td>1.021443e-13</td>\n",
"        <td>1.610692e-13</td>\n",
"        <td>-2.326883e-18</td>\n",
"        <td>-3.286722e-17</td>\n",
"        <td>1.861506e-17</td>\n",
"        <td>2018-10-14 12:32:49.986491648</td>\n",
"        <td>9.307531e-18</td>\n",
"        <td>9.307531e-18</td>\n",
"        <td>-2.792259e-17</td>\n",
"      </tr>\n",
"      <tr>\n",
"        <th>min</th>\n",
"        <td>-1.731876e+00</td>\n",
"        <td>-8.601704e-01</td>\n",
"        <td>-4.202515e+00</td>\n",
"        <td>-6.331382e+00</td>\n",
"        <td>-6.358088e-01</td>\n",
"        <td>-3.003286e-01</td>\n",
"        <td>-5.224105e-01</td>\n",
"        <td>2011-03-28 00:00:00</td>\n",
"        <td>-6.831252e-01</td>\n",
"        <td>-1.865162e-01</td>\n",
"        <td>-8.570911e-01</td>\n",
"      </tr>\n",
"      <tr>\n",
"        <th>25%</th>\n",
"        <td>-8.693035e-01</td>\n",
"        <td>-7.607565e-01</td>\n",
"        <td>-7.125107e-01</td>\n",
"        <td>-6.694202e-01</td>\n",
"        <td>-3.485840e-01</td>\n",
"        <td>-3.003286e-01</td>\n",
"        <td>-4.999636e-01</td>\n",

```

```

"      <td>2018-07-31 00:00:00</td>\n",
"      <td>-6.580822e-01</td>\n",
"      <td>-1.865162e-01</td>\n",
"      <td>-8.570911e-01</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>50%</th>\n",
"      <td>6.080368e-02</td>\n",
"      <td>-4.685692e-01</td>\n",
"      <td>-1.076743e-01</td>\n",
"      <td>-7.603542e-02</td>\n",
"      <td>-1.945649e-01</td>\n",
"      <td>-2.004263e-01</td>\n",
"      <td>-4.101757e-01</td>\n",
"      <td>2019-05-25 00:00:00</td>\n",
"      <td>-4.514775e-01</td>\n",
"      <td>-1.865162e-01</td>\n",
"      <td>-5.151708e-01</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>75%</th>\n",
"      <td>9.227443e-01</td>\n",
"      <td>5.062474e-01</td>\n",
"      <td>6.265930e-01</td>\n",
"      <td>3.442517e-01</td>\n",
"      <td>9.265994e-02</td>\n",
"      <td>-1.005239e-01</td>\n",
"      <td>1.631676e-02</td>\n",
"      <td>2019-06-24 00:00:00</td>\n",
"      <td>3.060730e-01</td>\n",
"      <td>-1.561803e-01</td>\n",
"      <td>8.677069e-01</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>max</th>\n",
"      <td>1.590116e+00</td>\n",
"      <td>2.628867e+00</td>\n",
"      <td>3.376645e+00</td>\n",
"      <td>5.181670e+00</td>\n",
"      <td>4.099098e+01</td>\n",
"      <td>6.208868e+01</td>\n",
"      <td>1.359673e+01</td>\n",
"      <td>2019-07-08 00:00:00</td>\n",
"      <td>3.594225e+01</td>\n",
"      <td>9.702981e+00</td>\n",
"      <td>1.916263e+00</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>std</th>\n",
"      <td>1.000010e+00</td>\n",
"      <td>1.000010e+00</td>\n",
"      <td>1.000010e+00</td>\n",
"      <td>1.000010e+00</td>\n",
"      <td>1.000010e+00</td>\n",
"      <td>1.000010e+00</td>\n",
"      <td>1.000010e+00</td>\n",
"      <td>NaN</td>\n",
"      <td>1.000010e+00</td>\n",
"      <td>1.000010e+00</td>\n",
"      <td>1.000010e+00</td>\n",
"    </tr>\n",
"  </tbody>\n",
"</table>\n",
"</div>"
],
"text/plain": [
  id      host_id      latitude      longitude      price

```

```

\\n",
"count  4.885800e+04  4.885800e+04  4.885800e+04  4.885800e+04  4.885800e+04
\n",
"mean   -1.489205e-16 -3.723012e-17  1.021443e-13  1.610692e-13 -2.326883e-18
\n",
"min    -1.731876e+00 -8.601704e-01 -4.202515e+00 -6.331382e+00 -6.358088e-01
\n",
"25%    -8.693035e-01 -7.607565e-01 -7.125107e-01 -6.694202e-01 -3.485840e-01
\n",
"50%     6.080368e-02 -4.685692e-01 -1.076743e-01 -7.603542e-02 -1.945649e-01
\n",
"75%     9.227443e-01  5.062474e-01  6.265930e-01  3.442517e-01  9.265994e-02
\n",
"max     1.590116e+00  2.628867e+00  3.376645e+00  5.181670e+00  4.099098e+01
\n",
"std     1.000010e+00  1.000010e+00  1.000010e+00  1.000010e+00  1.000010e+00
\n",
"\n",
"      minimum_nights  number_of_reviews  last_review  \\n",
"count  4.885800e+04  4.885800e+04  48858  \n",
"mean   -3.286722e-17  1.861506e-17  2018-10-14 12:32:49.986491648  \n",
"min    -3.003286e-01  -5.224105e-01  2011-03-28 00:00:00  \n",
"25%    -3.003286e-01  -4.999636e-01  2018-07-31 00:00:00  \n",
"50%    -2.004263e-01  -4.101757e-01  2019-05-25 00:00:00  \n",
"75%    -1.005239e-01  1.631676e-02  2019-06-24 00:00:00  \n",
"max     6.208868e+01  1.359673e+01  2019-07-08 00:00:00  \n",
"std     1.000010e+00  1.000010e+00  NaN  \n",
"\n",
"      reviews_per_month  calculated_host_listings_count  availability_365  \n",
"count  4.885800e+04  4.885800e+04  4.885800e+04  \n",
"mean   9.307531e-18  9.307531e-18  -2.792259e-17  \n",
"min    -6.831252e-01  -1.865162e-01  -8.570911e-01  \n",
"25%    -6.580822e-01  -1.865162e-01  -8.570911e-01  \n",
"50%    -4.514775e-01  -1.865162e-01  -5.151708e-01  \n",
"75%     3.060730e-01  -1.561803e-01  8.677069e-01  \n",
"max     3.594225e+01  9.702981e+00  1.916263e+00  \n",
"std     1.000010e+00  1.000010e+00  1.000010e+00  "
]
},
"execution_count": 221,
"metadata": {},
"output_type": "execute_result"
}
],
"source": [
"df.describe()"
]
},
{
"cell_type": "code",
"execution_count": 223,
"id": "0b2881b9-1670-47d6-baed-2caca440f568",
"metadata": {},
"outputs": [
{
"name": "stdout",
"output_type": "stream",
"text": [
"
      id  host_id  latitude  longitude  \\n",
"count  1.000000  0.588221 -0.003116  0.091076  \n",
"host_id  0.588221  1.000000  0.020193  0.127198  \n",
"latitude -0.003116  0.020193  1.000000  0.084819  \n",
"longitude 0.091076  0.127198  0.084819  1.000000  \n",
"price     0.010564  0.015328  0.033944 -0.149954  \n",
"minimum_nights -0.012038 -0.017027  0.025893 -0.062893  \n",
"number_of_reviews -0.320020 -0.140273 -0.015198  0.059151  \n",
"reviews_per_month 0.180095  0.209644 -0.018702  0.138759  \n",

```

```

"calculated_host_listings_count 0.133224 0.154954 0.019548 -0.114746 \n",
"availability_365 0.085616 0.203743 -0.010775 0.082754 \n",
"\n",
"
price minimum_nights number_of_reviews
\\n",
"id 0.010564 -0.012038 -0.320020 \n",
"host_id 0.015328 -0.017027 -0.140273 \n",
"latitude 0.033944 0.025893 -0.015198 \n",
"longitude -0.149954 -0.062893 0.059151 \n",
"price 1.000000 0.042804 -0.047949 \n",
"minimum_nights 0.042804 1.000000 -0.081610 \n",
"number_of_reviews -0.047949 -0.081610 1.000000 \n",
"reviews_per_month -0.050586 -0.127378 0.589221 \n",
"calculated_host_listings_count 0.057460 0.131313 -0.072408 \n",
"availability_365 0.081817 0.145953 0.171855 \n",
"\n",
"
reviews_per_month \\n",
"id 0.180095 \n",
"host_id 0.209644 \n",
"latitude -0.018702 \n",
"longitude 0.138759 \n",
"price -0.050586 \n",
"minimum_nights -0.127378 \n",
"number_of_reviews 0.589221 \n",
"reviews_per_month 1.000000 \n",
"calculated_host_listings_count -0.047366 \n",
"availability_365 0.163751 \n",
"\n",
"
calculated_host_listings_count \\n",
"id 0.133224 \n",
"host_id 0.154954 \n",
"latitude 0.019548 \n",
"longitude -0.114746 \n",
"price 0.057460 \n",
"minimum_nights 0.131313 \n",
"number_of_reviews -0.072408 \n",
"reviews_per_month -0.047366 \n",
"calculated_host_listings_count 1.000000 \n",
"availability_365 0.225784 \n",
"\n",
"
availability_365 \n",
"id 0.085616 \n",
"host_id 0.203743 \n",
"latitude -0.010775 \n",
"longitude 0.082754 \n",
"price 0.081817 \n",
"minimum_nights 0.145953 \n",
"number_of_reviews 0.171855 \n",
"reviews_per_month 0.163751 \n",
"calculated_host_listings_count 0.225784 \n",
"availability_365 1.000000 \n"
]
}
],
"source": [
"numeric_df = df.select_dtypes(include=['float64', 'int64'])\n",
"corr_matrix = numeric_df.corr()\n",
"\n",
"print(corr_matrix)"
]
},
{
"cell_type": "markdown",
"id": "567249dc-3989-4218-9de8-a7ef5dbc2c97",
"metadata": {},
"source": [
"### Finding unique values count"
]
}

```

```

]
},
{
  "cell_type": "code",
  "execution_count": 226,
  "id": "bef3f168-7d8c-4da0-8cfc-5928a9c29c85",
  "metadata": {},
  "outputs": [
    {
      "data": {
        "text/plain": [
          "id          48858\n",
          "name        47884\n",
          "host_id     37425\n",
          "host_name   11450\n",
          "neighbourhood_group  5\n",
          "neighbourhood 221\n",
          "latitude    19039\n",
          "longitude   14716\n",
          "room_type   3\n",
          "price       674\n",
          "minimum_nights 108\n",
          "number_of_reviews 394\n",
          "last_review 1764\n",
          "reviews_per_month 938\n",
          "calculated_host_listings_count 47\n",
          "availability_365 366\n",
          "dtype: int64"
        ]
      },
      "execution_count": 226,
      "metadata": {},
      "output_type": "execute_result"
    }
  ],
  "source": [
    "df.nunique()"
  ]
},
{
  "cell_type": "code",
  "execution_count": 228,
  "id": "cbabf200-b9d6-4882-b752-e8113c0f8273",
  "metadata": {},
  "outputs": [
    {
      "data": {
        "text/html": [
          "<div>\n",
          "<style scoped>\n",
          "  .dataframe tbody tr th:only-of-type {\n",
          "    vertical-align: middle;\n",
          "  }\n",
          "\n",
          "  .dataframe tbody tr th {\n",
          "    vertical-align: top;\n",
          "  }\n",
          "\n",
          "  .dataframe thead th {\n",
          "    text-align: right;\n",
          "  }\n",
          "</style>\n",
          "<table border='1' class='dataframe'>\n",
          "  <thead>\n",
          "    <tr style='text-align: right;'>\n",
          "      <th></th>\n",
          "      <th>id</th>\n",

```

```

"      <th>name</th>\n",
"      <th>host_id</th>\n",
"      <th>host_name</th>\n",
"      <th>neighbourhood</th>\n",
"      <th>latitude</th>\n",
"      <th>longitude</th>\n",
"      <th>room_type</th>\n",
"      <th>price</th>\n",
"      <th>minimum_nights</th>\n",
"      <th>number_of_reviews</th>\n",
"      <th>last_review</th>\n",
"      <th>reviews_per_month</th>\n",
"      <th>calculated_host_listings_count</th>\n",
"      <th>availability_365</th>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>neighbourhood_group</th>\n",
"      <th></th>\n",
"      <th></th>\n",
"      <th></th>\n",
"      <th></th>\n",
"      <th></th>\n",
"      <th></th>\n",
"      <th></th>\n",
"      <th></th>\n",
"      <th></th>\n",
"      <th></th>\n",
"      <th></th>\n",
"      <th></th>\n",
"      <th></th>\n",
"      <th></th>\n",
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"    <tr>\n",
"      <th>bronx</th>\n",
"      <td>1089</td>\n",
"      <td>1089</td>\n",
"      <td>1089</td>\n",
"      <td>1089</td>\n",
"      <td>1089</td>\n",
"      <td>1089</td>\n",
"      <td>1089</td>\n",
"      <td>1089</td>\n",
"      <td>1089</td>\n",
"      <td>1089</td>\n",
"      <td>1089</td>\n",
"      <td>1089</td>\n",
"      <td>1089</td>\n",
"      <td>1089</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>brooklyn</th>\n",
"      <td>20089</td>\n",
"      <td>20089</td>\n",
"      <td>20089</td>\n",
"      <td>20089</td>\n",
"      <td>20089</td>\n",
"      <td>20089</td>\n",
"      <td>20089</td>\n",
"      <td>20089</td>\n",
"      <td>20089</td>\n",
"      <td>20089</td>\n",
"      <td>20089</td>\n",
"      <td>20089</td>\n",
"      <td>20089</td>\n",
"      <td>20089</td>\n",

```

```

"      <td>20089</td>\n",
"      <td>20089</td>\n",
"      <td>20089</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>manhattan</th>\n",
"      <td>21643</td>\n",
"      <td>21643</td>\n",
"      <td>21643</td>\n",
"      <td>21643</td>\n",
"      <td>21643</td>\n",
"      <td>21643</td>\n",
"      <td>21643</td>\n",
"      <td>21643</td>\n",
"      <td>21643</td>\n",
"      <td>21643</td>\n",
"      <td>21643</td>\n",
"      <td>21643</td>\n",
"      <td>21643</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>queens</th>\n",
"      <td>5664</td>\n",
"      <td>5664</td>\n",
"      <td>5664</td>\n",
"      <td>5664</td>\n",
"      <td>5664</td>\n",
"      <td>5664</td>\n",
"      <td>5664</td>\n",
"      <td>5664</td>\n",
"      <td>5664</td>\n",
"      <td>5664</td>\n",
"      <td>5664</td>\n",
"      <td>5664</td>\n",
"      <td>5664</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>staten island</th>\n",
"      <td>373</td>\n",
"      <td>373</td>\n",
"      <td>373</td>\n",
"      <td>373</td>\n",
"      <td>373</td>\n",
"      <td>373</td>\n",
"      <td>373</td>\n",
"      <td>373</td>\n",
"      <td>373</td>\n",
"      <td>373</td>\n",
"      <td>373</td>\n",
"      <td>373</td>\n",
"      <td>373</td>\n",
"      <td>373</td>\n",
"    </tr>\n",
"  </tbody>\n",
"</table>\n",
"</div>"
],
"text/plain": [
"      id      name      host_id      host_name      neighbourhood      \\\n",
"neighbourhood_group      \n",
"bronx      1089      1089      1089      1089      1089      \n",
"brooklyn      20089      20089      20089      20089      20089      \n",

```



```

"manhattan          21643  21643    21643    21643          21643  \n",
"queens              5664   5664    5664    5664          5664  \n",
"staten island       373    373     373     373          373  \n",
"\n",
"                    latitude longitude room_type price minimum_nights
\\n",
"neighbourhood_group\n",
"bronx              1089      1089      1089    1089          1089  \n",
"brooklyn           20089     20089     20089    20089          20089  \n",
"manhattan          21643     21643     21643    21643          21643  \n",
"queens              5664     5664     5664    5664          5664  \n",
"staten island       373      373      373     373          373  \n",
"\n",
"                    number_of_reviews last_review reviews_per_month \\n",
"neighbourhood_group\n",
"bronx              1089      1089          1089  \n",
"brooklyn           20089     20089          20089  \n",
"manhattan          21643     21643          21643  \n",
"queens              5664     5664          5664  \n",
"staten island       373      373          373  \n",
"\n",
"                    calculated_host_listings_count availability_365 \n",
"neighbourhood_group\n",
"bronx              1089          1089  \n",
"brooklyn           20089          20089  \n",
"manhattan          21643          21643  \n",
"queens              5664          5664  \n",
"staten island       373          373  "
]
},
"execution_count": 228,
"metadata": {},
"output_type": "execute_result"
}
],
"source": [
"neigh_grp_count = df.groupby(\"neighbourhood_group\").count().head()\n",
"neigh_grp_count"
]
},
{
"cell_type": "markdown",
"id": "69eb0dbc-92a4-4bc1-b6ff-c13221d92dac",
"metadata": {},
"source": [
"### Finding and removing outliers"
]
},
{
"cell_type": "code",
"execution_count": 231,
"id": "5f325371-8b63-44a3-aada-11ba7c8c441b",
"metadata": {},
"outputs": [],
"source": [
"Q1 = df['price'].quantile(0.25)\n",
"Q3 = df['price'].quantile(0.75)\n",
"IQR = Q3 - Q1\n",
"lower_bound = Q1 - 1.5 * IQR\n",
"upper_bound = Q3 + 1.5 * IQR\n",
"\n",
"df_cleaned = df[(df['price'] >= lower_bound) & (df['price'] <= upper_bound)]"
]
},
{
"cell_type": "markdown",
"id": "9158c6b2-9aac-4da5-8ad1-cd23c4a1e63f",

```

```

"metadata": {},
"source": [
    "### StandardScaler only numeric columns \n",
    "\n",
    "##### StandardScaler(): This calculates the mean and standard deviation of each
feature and scales the data accordingly."
]
},
{
    "cell_type": "code",
    "execution_count": 234,
    "id": "14a6e8ba-93f5-460f-8b52-d503d6254055",
    "metadata": {},
    "outputs": [
        {
            "name": "stdout",
            "output_type": "stream",
            "text": [
                "
                id
                name
                host_id
\\n",
                "0
                -1.731876
                Clean & quiet apt home by the park
                -0.860166
\\n",
                "1
                -1.731871
                Skylit Midtown Castle
                -0.860165
\\n",
                "2
                -1.731775
                THE VILLAGE OF HARLEM....NEW YORK !
                -0.860142
\\n",
                "3
                -1.731758
                Cozy Entire Floor of Brownstone
                -0.860139
\\n",
                "4
                -1.731650
                Entire Apt: Spacious Studio/Loft by central park
                -0.860110
\\n",
                "...
                ...
                ...
                ...
\\n",
                "48890
                1.589881
                Charming one bedroom - newly renovated rowhouse
                -0.755494
\\n",
                "48891
                1.589917
                Affordable room in Bushwick/East Williamsburg
                -0.776630
\\n",
                "48892
                1.589951
                Sunny Studio at Historical Neighborhood
                -0.561397
\\n",
                "48893
                1.589967
                43rd St. Time Square-cozy single bed
                -0.466096
\\n",
                "48894
                1.590116
                Trendy duplex in the very heart of Hell's Kitchen
                0.006208
\\n",
                "\n",
                "
                host_name
                neighbourhood_group
                neighbourhood
                latitude
                \n",
                "0
                John
                brooklyn
                Kensington
                -1.493773
                \n",
                "1
                Jennifer
                manhattan
                Midtown
                0.452597
                \n",
                "2
                Elisabeth
                manhattan
                Harlem
                1.468605
                \n",
                "3
                LisaRoxanne
                brooklyn
                Clinton Hill
                -0.803291
                \n",
                "4
                Laura
                manhattan
                East Harlem
                1.275857
                \n",
                "...
                ...
                ...
                ...
                \n",
                "48890
                Sabrina
                brooklyn
                Bedford-Stuyvesant
                -0.924515
                \n",
                "48891
                Marisol
                brooklyn
                Bushwick
                -0.497022
                \n",
                "48892
                Ilgar & Aysel
                manhattan
                Harlem
                1.573690
                \n",
                "48893
                Taz
                manhattan
                Hell's Kitchen
                0.523938
                \n",
                "48894
                Christophe
                manhattan
                Hell's Kitchen
                0.643695
                \n",
                "\n",
                "
                longitude
                room_type
                price
                minimum_nights
                \n",
                "0
                -0.437612
                private room
                -0.015570
                -0.300329
                \n",
                "1
                -0.684585
                entire home/apt
                0.300794
                -0.300329
                \n",
                "2
                0.222498
                private room
                -0.011407
                -0.200426
                \n",
                "3
                -0.164426
                entire home/apt
                -0.265330
                -0.300329
                \n",
                "4
                0.177220
                entire home/apt
                -0.302795
                0.149232
                \n",
                "...
                ...
                ...
                ...
                \n",
                "48890
                0.048101
                private room
                -0.344421
                -0.250377
                \n",
                "48891
                0.411628
                private room
                -0.469302
                -0.150475
                \n",
                "48892
                0.075831
                entire home/apt
                -0.157101
                0.149232
                \n",
                "48893
                -0.843818
                shared room
                -0.406862
                -0.300329
                \n"
            ]
        }
    ]
}

```

```

"48894    -0.805039    private room -0.261168    -0.000622    \n",
"\n",
"    number_of_reviews last_review reviews_per_month \\n",
"0        -0.320388    2018-10-19    -0.551649    \n",
"1         0.487703    2019-05-21    -0.445217    \n",
"2        -0.522411    2019-05-21    -0.683125    \n",
"3         5.538271    2019-07-05     2.221862    \n",
"4        -0.320388    2018-11-19    -0.620518    \n",
"...      ...      ...      ...      \n",
"48890        -0.522411    2019-07-08    -0.683125    \n",
"48891        -0.522411    2019-07-08    -0.683125    \n",
"48892        -0.522411    2019-07-08    -0.683125    \n",
"48893        -0.522411    2019-07-08    -0.683125    \n",
"48894        -0.522411    2019-07-08    -0.683125    \n",
"\n",
"    calculated_host_listings_count availability_365 \n",
"0                -0.034837     1.916263 \n",
"1                -0.156180     1.840280 \n",
"2                -0.186516     1.916263 \n",
"3                -0.186516     0.616965 \n",
"4                -0.186516    -0.857091 \n",
"...      ...      ...      \n",
"48890                -0.156180    -0.788707 \n",
"48891                -0.156180    -0.583555 \n",
"48892                -0.186516    -0.651939 \n",
"48893                -0.034837    -0.841895 \n",
"48894                -0.186516    -0.682332 \n",
"\n",
"[48858 rows x 16 columns]\n"
]
},
{
"source": [
"from sklearn.preprocessing import StandardScaler\n",
"\n",
"numeric_cols = df.select_dtypes(include=['float64', 'int64']).columns\n",
"scaler = StandardScaler()\n",
"df[numeric_cols] = scaler.fit_transform(df[numeric_cols])\n",
"print(df)"
]
},
{
"cell_type": "code",
"execution_count": 236,
"id": "68484c43-ddf7-43d3-9d85-a237b0d0bb76",
"metadata": {},
"outputs": [
{
"data": {
"text/plain": [
"0        False\n",
"1        False\n",
"2        False\n",
"3        False\n",
"4        False\n",
"...      \n",
"48890     False\n",
"48891     False\n",
"48892     False\n",
"48893     False\n",
"48894     False\n",
"Name: name, Length: 48858, dtype: bool"
]
}
},
{
"execution_count": 236,
"metadata": {},
"output_type": "execute_result"
}
]
}

```

```

}
],
"source": [
  "duplicates = df['name'].duplicated()\n",
  "duplicates"
]
},
{
  "cell_type": "code",
  "execution_count": 238,
  "id": "c5bf56dd-b8fa-48e8-939c-5da06fa9c5f4",
  "metadata": {},
  "outputs": [
    {
      "name": "stdout",
      "output_type": "stream",
      "text": [
        "
          id      host_id  latitude  longitude  \\\n",
        "id      1.000000  0.588221 -0.003116   0.091076   \n",
        "host_id  0.588221  1.000000  0.020193   0.127198   \n",
        "latitude -0.003116  0.020193  1.000000   0.084819   \n",
        "longitude 0.091076  0.127198  0.084819   1.000000   \n",
        "price     0.010564  0.015328  0.033944  -0.149954   \n",
        "minimum_nights -0.012038 -0.017027  0.025893  -0.062893   \n",
        "number_of_reviews -0.320020 -0.140273 -0.015198   0.059151   \n",
        "reviews_per_month 0.180095  0.209644 -0.018702   0.138759   \n",
        "calculated_host_listings_count 0.133224  0.154954  0.019548  -0.114746   \n",
        "availability_365 0.085616  0.203743 -0.010775   0.082754   \n",
        "\n",
        "
          price  minimum_nights  number_of_reviews
\\\n",
        "id      0.010564      -0.012038      -0.320020   \n",
        "host_id 0.015328      -0.017027      -0.140273   \n",
        "latitude 0.033944      0.025893      -0.015198   \n",
        "longitude -0.149954      -0.062893      0.059151   \n",
        "price     1.000000      0.042804      -0.047949   \n",
        "minimum_nights 0.042804      1.000000      -0.081610   \n",
        "number_of_reviews -0.047949      -0.081610      1.000000   \n",
        "reviews_per_month -0.050586      -0.127378      0.589221   \n",
        "calculated_host_listings_count 0.057460      0.131313      -0.072408   \n",
        "availability_365 0.081817      0.145953      0.171855   \n",
        "\n",
        "
          reviews_per_month  \\\n",
        "id      0.180095   \n",
        "host_id 0.209644   \n",
        "latitude -0.018702   \n",
        "longitude 0.138759   \n",
        "price     -0.050586   \n",
        "minimum_nights -0.127378   \n",
        "number_of_reviews 0.589221   \n",
        "reviews_per_month 1.000000   \n",
        "calculated_host_listings_count -0.047366   \n",
        "availability_365 0.163751   \n",
        "\n",
        "
          calculated_host_listings_count  \\\n",
        "id      0.133224   \n",
        "host_id 0.154954   \n",
        "latitude 0.019548   \n",
        "longitude -0.114746   \n",
        "price     0.057460   \n",
        "minimum_nights 0.131313   \n",
        "number_of_reviews -0.072408   \n",
        "reviews_per_month -0.047366   \n",
        "calculated_host_listings_count 1.000000   \n",
        "availability_365 0.225784   \n",
        "\n",
        "
          availability_365   \n",

```

```

        "id"                0.085616  \n",
        "host_id"           0.203743  \n",
        "latitude"          -0.010775 \n",
        "longitude"         0.082754  \n",
        "price"             0.081817  \n",
        "minimum_nights"    0.145953  \n",
        "number_of_reviews" 0.171855  \n",
        "reviews_per_month" 0.163751  \n",
        "calculated_host_listings_count" 0.225784 \n",
        "availability_365"  1.000000  \n"
    ]
}
],
"source": [
    "numeric_df = df.select_dtypes(include=['float64', 'int64'])\n",
    "corr_matrix = numeric_df.corr()\n",
    "\n",
    "print(corr_matrix)"
]
},
{
    "cell_type": "code",
    "execution_count": 240,
    "id": "3a179b40-e643-48b0-932a-039e388370f0",
    "metadata": {},
    "outputs": [],
    "source": [
        "df.to_csv(\"cleaned_data.csv\")"
    ]
},
{
    "cell_type": "code",
    "execution_count": null,
    "id": "b903a359-9719-48bc-a705-39fd3c17ff1d",
    "metadata": {},
    "outputs": [],
    "source": []
}
],
"metadata": {
    "kernelspec": {
        "display_name": "Python 3 (ipykernel)",
        "language": "python",
        "name": "python3"
    },
    "language_info": {
        "codemirror_mode": {
            "name": "ipython",
            "version": 3
        },
        "file_extension": ".py",
        "mimetype": "text/x-python",
        "name": "python",
        "nbconvert_exporter": "python",
        "pygments_lexer": "ipython3",
        "version": "3.11.7"
    }
},
"nbformat": 4,
"nbformat_minor": 5
}

```