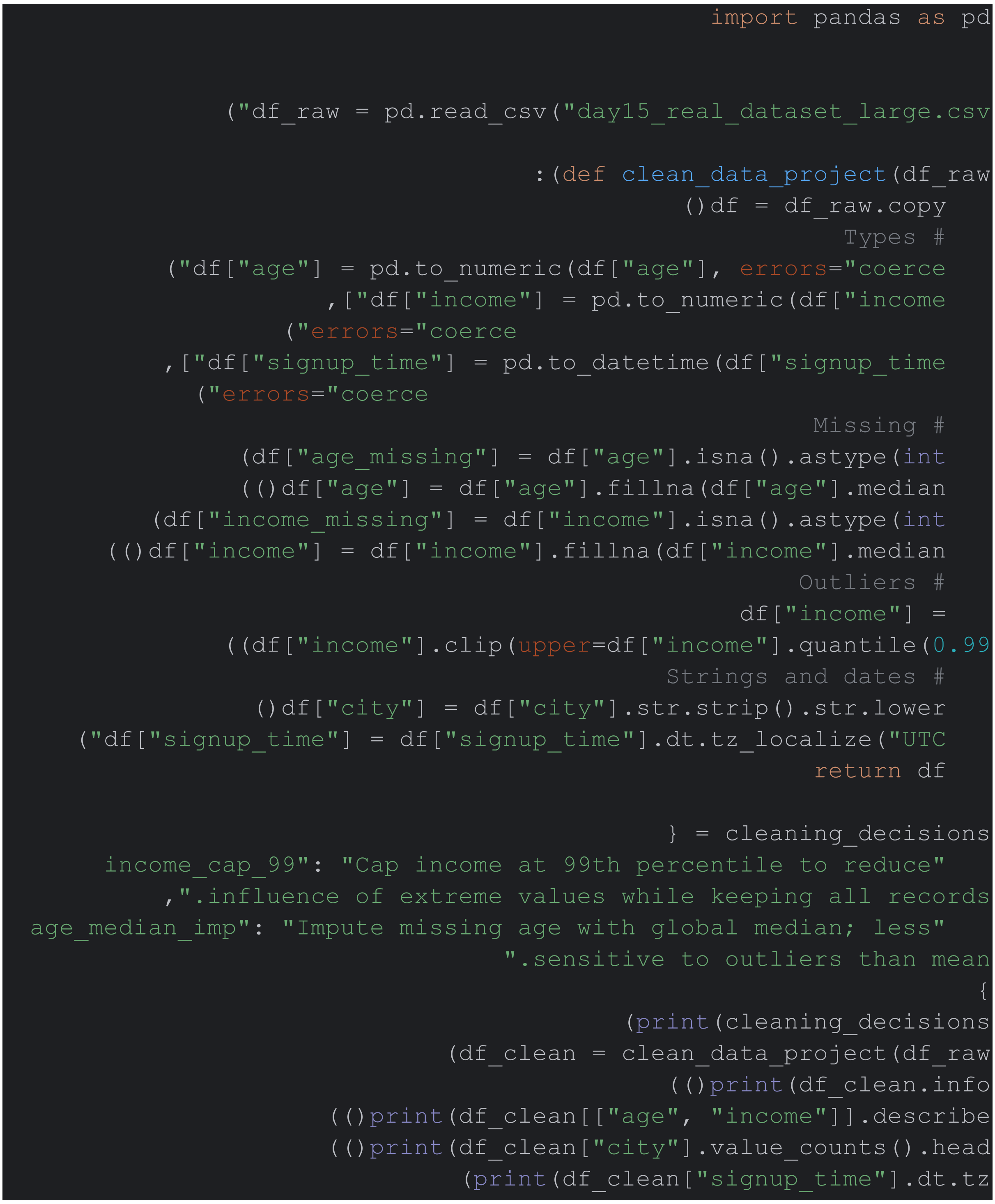

```
                                              import pandas as pd


             ("df_raw = pd.read_csv("day15_real_dataset_large.csv

                                  :(def clean_data_project(df_raw
                                            ()df = df_raw.copy
                                                       Types #
         ("df["age"] = pd.to_numeric(df["age"], errors="coerce
                    ,["df["income"] = pd.to_numeric(df["income
                 ("errors="coerce
         ,["df["signup_time"] = pd.to_datetime(df["signup_time
           ("errors="coerce
                                                     Missing #
              (df["age_missing"] = df["age"].isna().astype(int
              (()df["age"] = df["age"].fillna(df["age"].median
        (df["income_missing"] = df["income"].isna().astype(int
     (()df["income"] = df["income"].fillna(df["income"].median
                                                    Outliers #
                                                df["income"] =
             ((df["income"].clip(upper=df["income"].quantile(0.99
                                           Strings and dates #
               ()df["city"] = df["city"].str.strip().str.lower
   ("df["signup_time"] = df["signup_time"].dt.tz_localize("UTC
                                                     return df

                                           } = cleaning_decisions
     income_cap_99": "Cap income at 99th percentile to reduce"
         ,".influence of extreme values while keeping all records
age_median_imp": "Impute missing age with global median; less"
                                ".sensitive to outliers than mean
                                                                {
                                        (print(cleaning_decisions
                            (df_clean = clean_data_project(df_raw
                                           (()print(df_clean.info
                    (()print(df_clean[["age", "income"]].describe
                    (()print(df_clean["city"].value_counts().head
                             (print(df_clean["signup_time"].dt.tz
```