Your Fintastic Insights for TravelGuard

“Making Waves in Safety”

1. TEAM “VALUE COUNTS ! ”

ADAM

ADRIANA

SMITA

GLORIA

2. PROJECT OVERVIEW

1. ORIGINAL DATASET

⇒ list of documented shark attacks in media

⇒ Lots of missing data, inconsistent naming (activities, injuries), null values

2. OUR POSITION

⇒ Data research for a Travel Insurance Company

Reveal patterns/trends of shark attacks for an actuary team to adapt their pricing/offering based on the risk profile

⇒ OBJECTIVES

- Find the most risky profile within the top 20 countries

3. DATA CLEANING

⇒ removing unimportant information

⇒ removing NaNs

⇒ grouping key columns

⇒ date formatting

2. PROJECT OVERVIEW

4. EDA

⇒ **Analyzing key data:**

- gender
- age groups
- countries
- activities
- injuries

⇒ **Methods:**

- grouping
- pivot tables
- value counts

5. VISUALIZATION

⇒ **plotting charts to visualize the findings**

- time series
- pie charts
- bar charts

3. DATA CLEANING

CHALLENGES

⇒ planning

⇒a lot of missing values

⇒ mostly categorical data

⇒ date formatting

SOLUTIONS

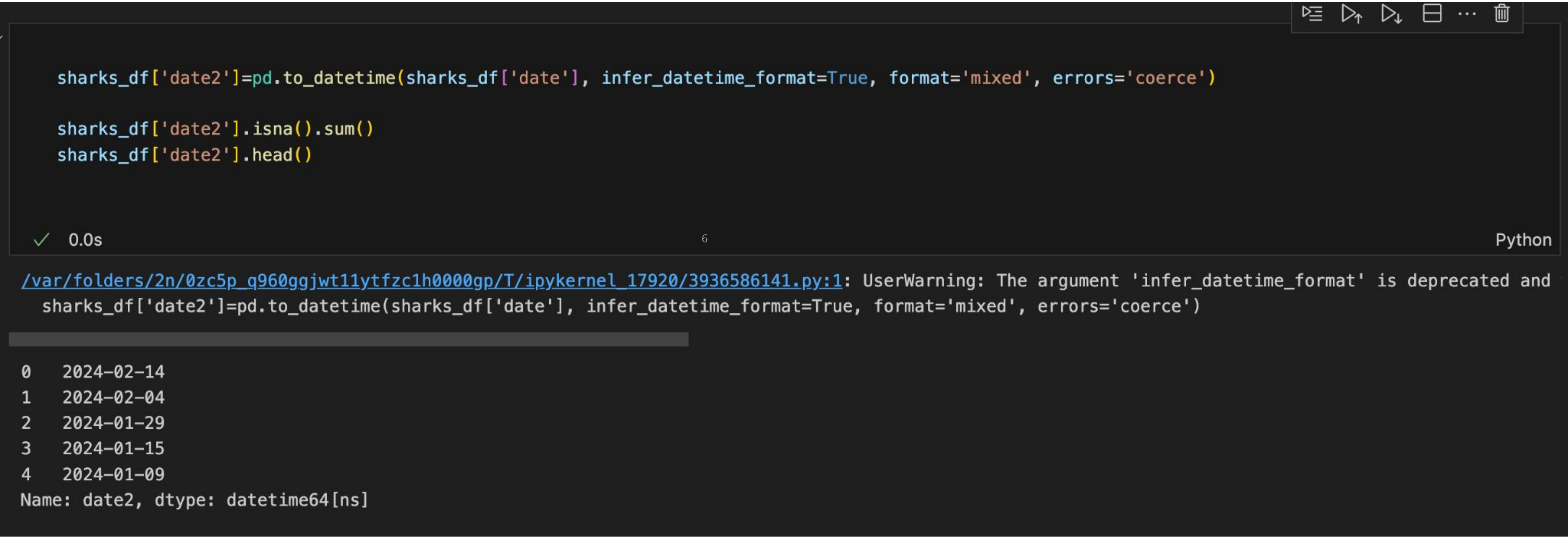
⇒ standardizing format of columns names

⇒date formatting

⇒ functions to group age, types of activities and injuries

⇒ dropping NaNs, removing unnecessary columns

3. DATE FORMATTING



```python
sharks_df['date2']=pd.to_datetime(sharks_df['date'], infer_datetime_format=True, format='mixed', errors='coerce')

sharks_df['date2'].isna().sum()
sharks_df['date2'].head()
```

```
/var/folders/2n/0zc5p_q960ggjwt11ytfzc1h0000gp/T/ipykernel_17920/3936586141.py:1: UserWarning: The argument 'infer_datetime_format' is deprecated and
  sharks_df['date2']=pd.to_datetime(sharks_df['date'], infer_datetime_format=True, format='mixed', errors='coerce')
```

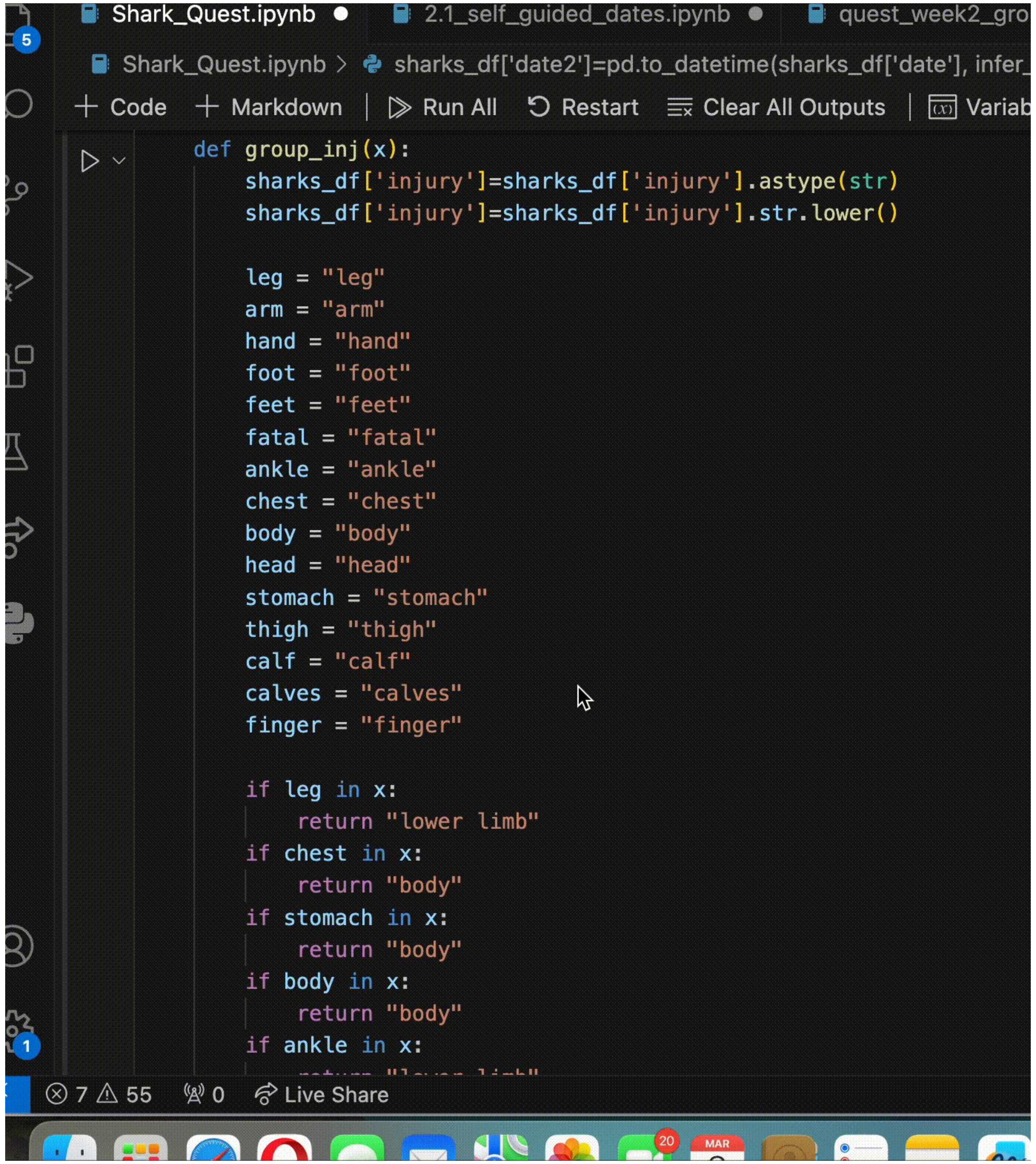
```
0   2024-02-14
1   2024-02-04
2   2024-01-29
3   2024-01-15
4   2024-01-09
Name: date2, dtype: datetime64[ns]
```

3. GROUPING

⇒ define parts of the body, activities, age groups etc.

⇒ define functions classify cases

⇒ apply functions create new columns with groups

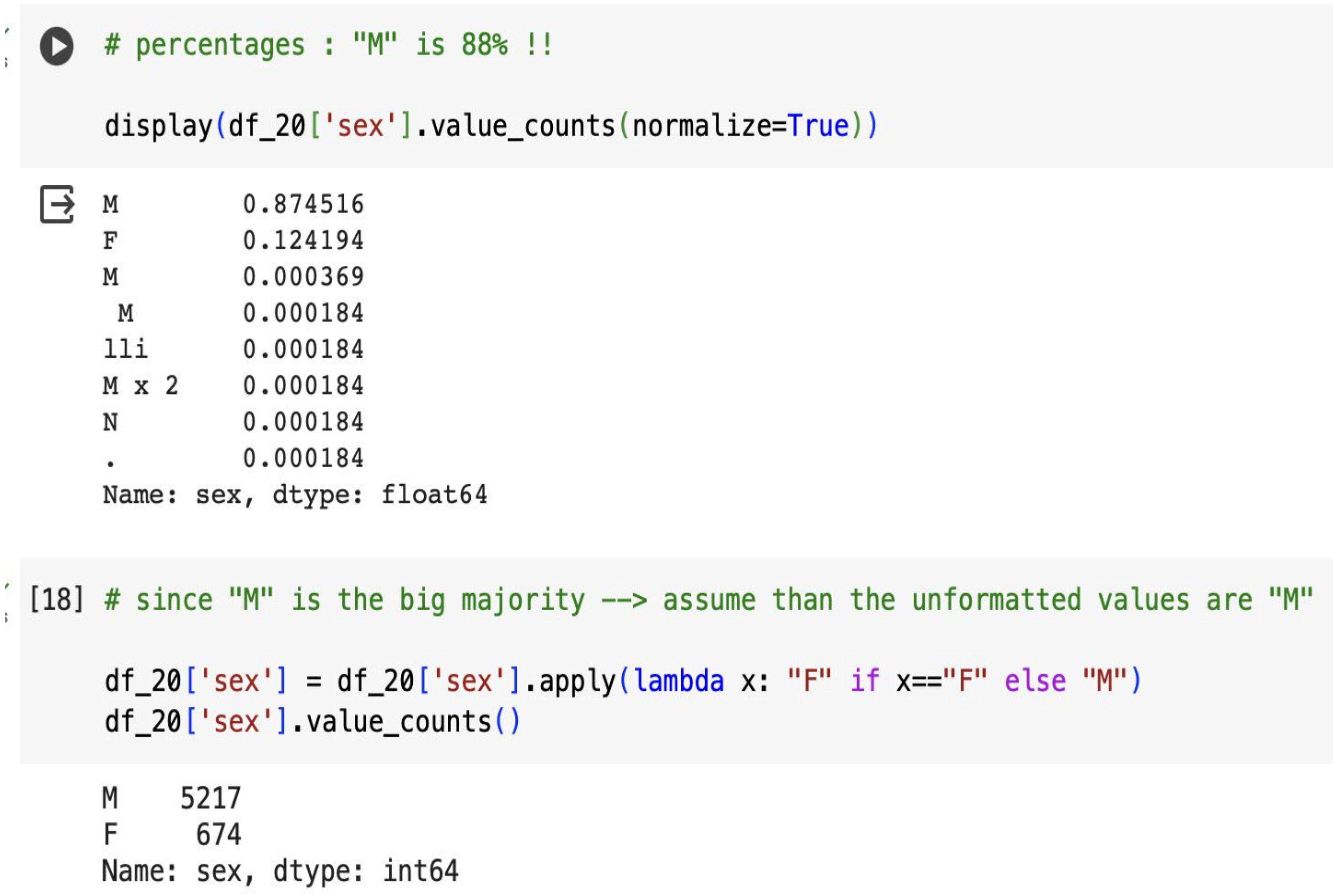

Shark_Quest.ipynb
2.1_self_guided_dates.ipynb
quest_week2_gro
Shark_Quest.ipynb > sharks_df['date2']=pd.to_datetime(sharks_df['date'], infer_
+ Code
+ Markdown
Run All
Restart
Clear All Outputs
Variab
7 55
0
Live Share

```
def group_inj(x):
    sharks_df['injury']=sharks_df['injury'].astype(str)
    sharks_df['injury']=sharks_df['injury'].str.lower()

    leg = "leg"
    arm = "arm"
    hand = "hand"
    foot = "foot"
    feet = "feet"
    fatal = "fatal"
    ankle = "ankle"
    chest = "chest"
    body = "body"
    head = "head"
    stomach = "stomach"
    thigh = "thigh"
    calf = "calf"
    calves = "calves"
    finger = "finger"

    if leg in x:
        return "lower limb"
    if chest in x:
        return "body"
    if stomach in x:
        return "body"
    if body in x:
        return "body"
    if ankle in x:
        return "lower limb"
```

3. FILLING EMPTY VALUES

⇒ Due to high number of NaNs we were only able to fill in missing values in gender

⇒ Most of the cases were men so we replaced missing or wrong values with men



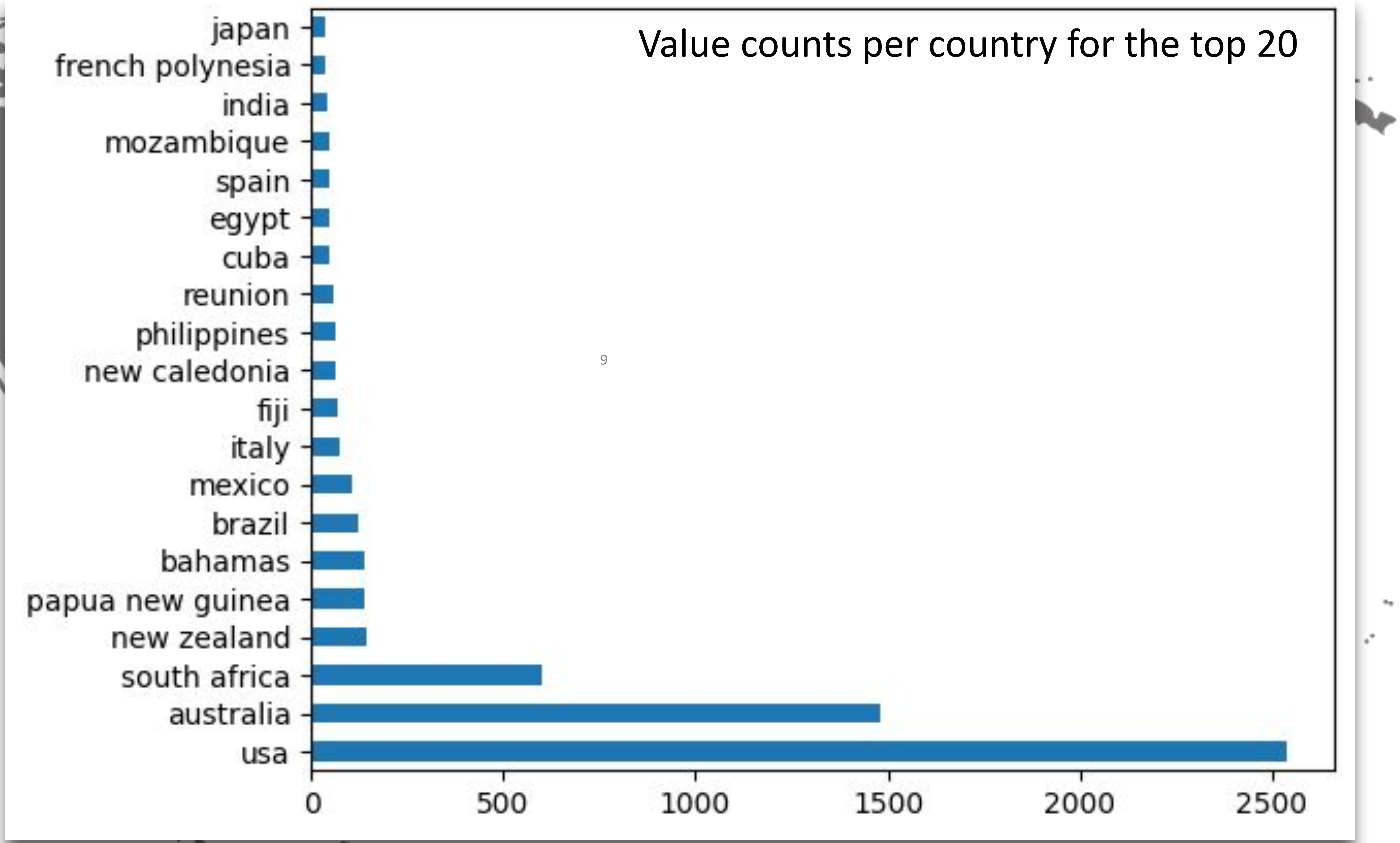
```
# percentages : "M" is 88% !!

display(df_20['sex'].value_counts(normalize=True))
```

```
M          0.874516
F          0.124194
M          0.000369
 M         0.000184
lli        0.000184
M x 2      0.000184
N          0.000184
.          0.000184
Name: sex, dtype: float64
```

```
[18] # since "M" is the big majority --> assume than the unformatted values are "M"

df_20['sex'] = df_20['sex'].apply(lambda x: "F" if x=="F" else "M")
df_20['sex'].value_counts()
```

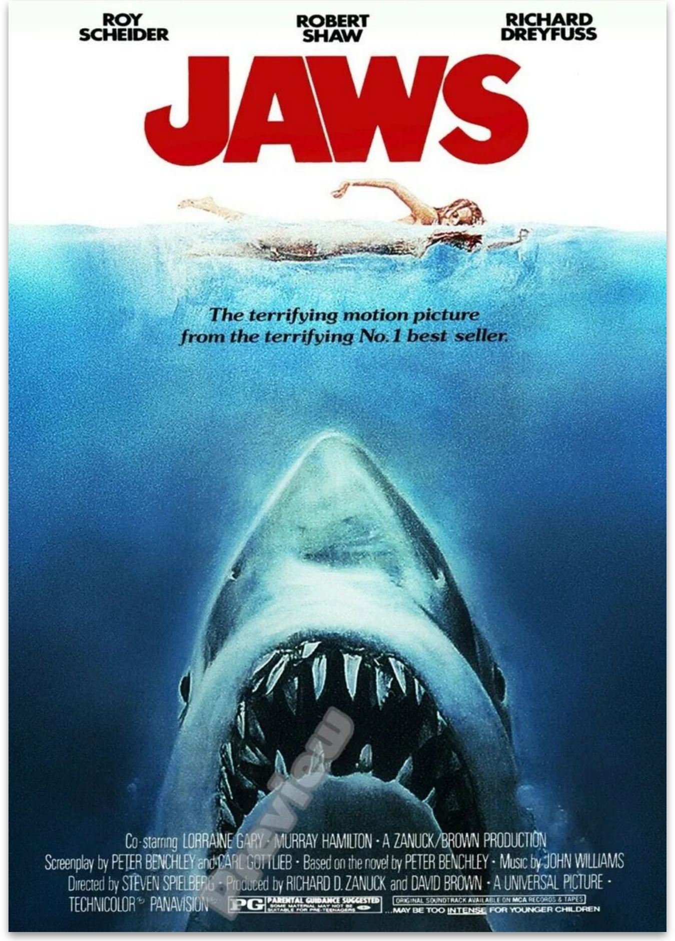
```
M    5217
F     674
Name: sex, dtype: int64
```

4. EXPLORATORY DATA ANALYSIS



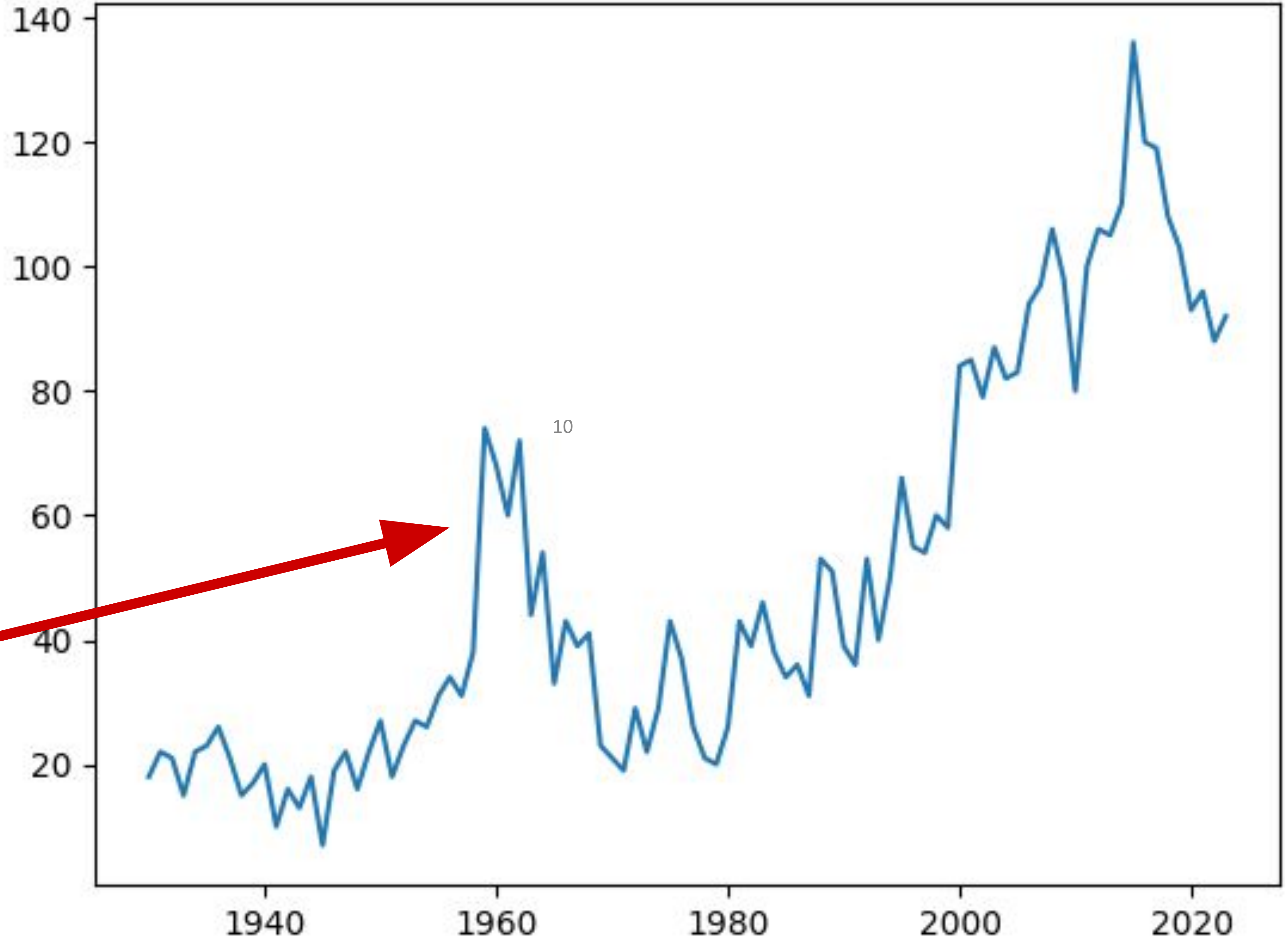
Value counts per country for the top 20

4. EXPLORATORY DATA ANALYSIS

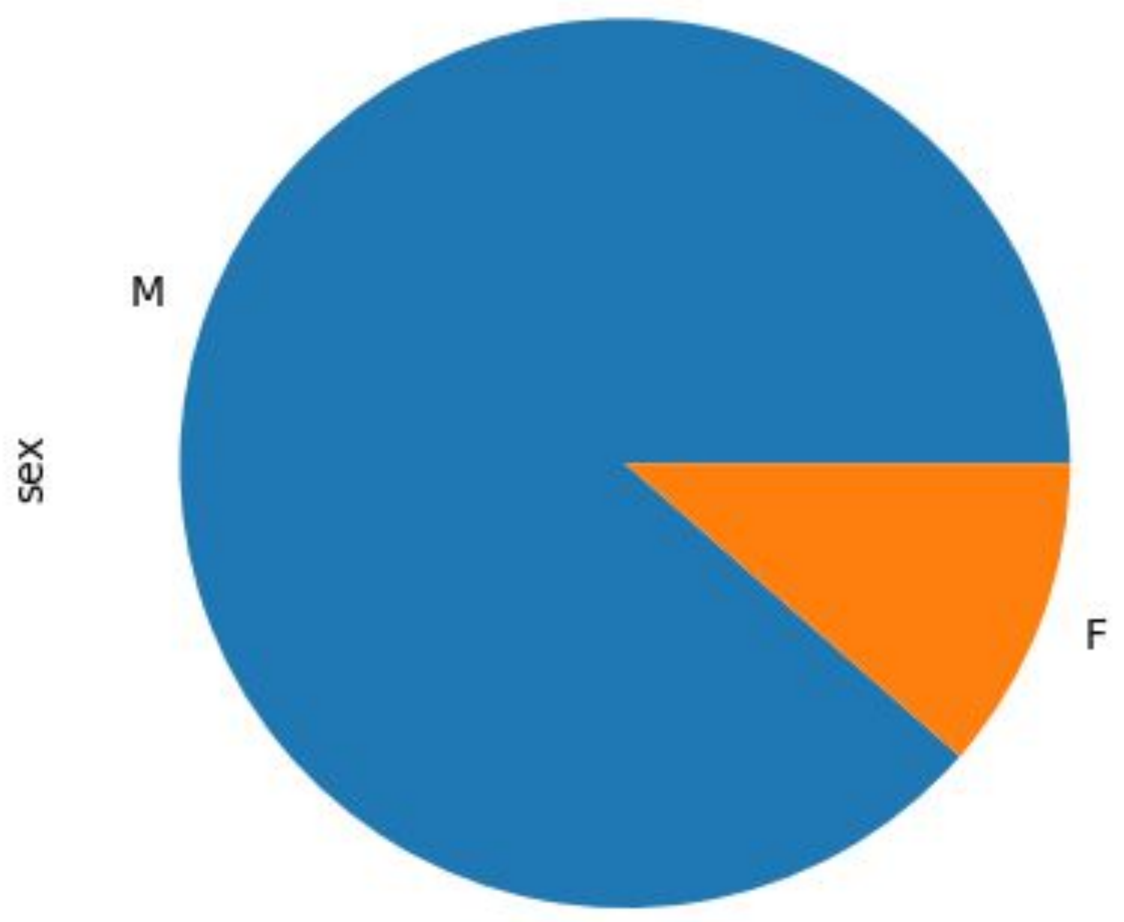

ROY SCHEIDER
ROBERT SHAW
RICHARD DREYFUSS
JAWS
The terrifying motion picture
from the terrifying No. 1 best seller.
PG

“Jaws” movie was released in 1975 …

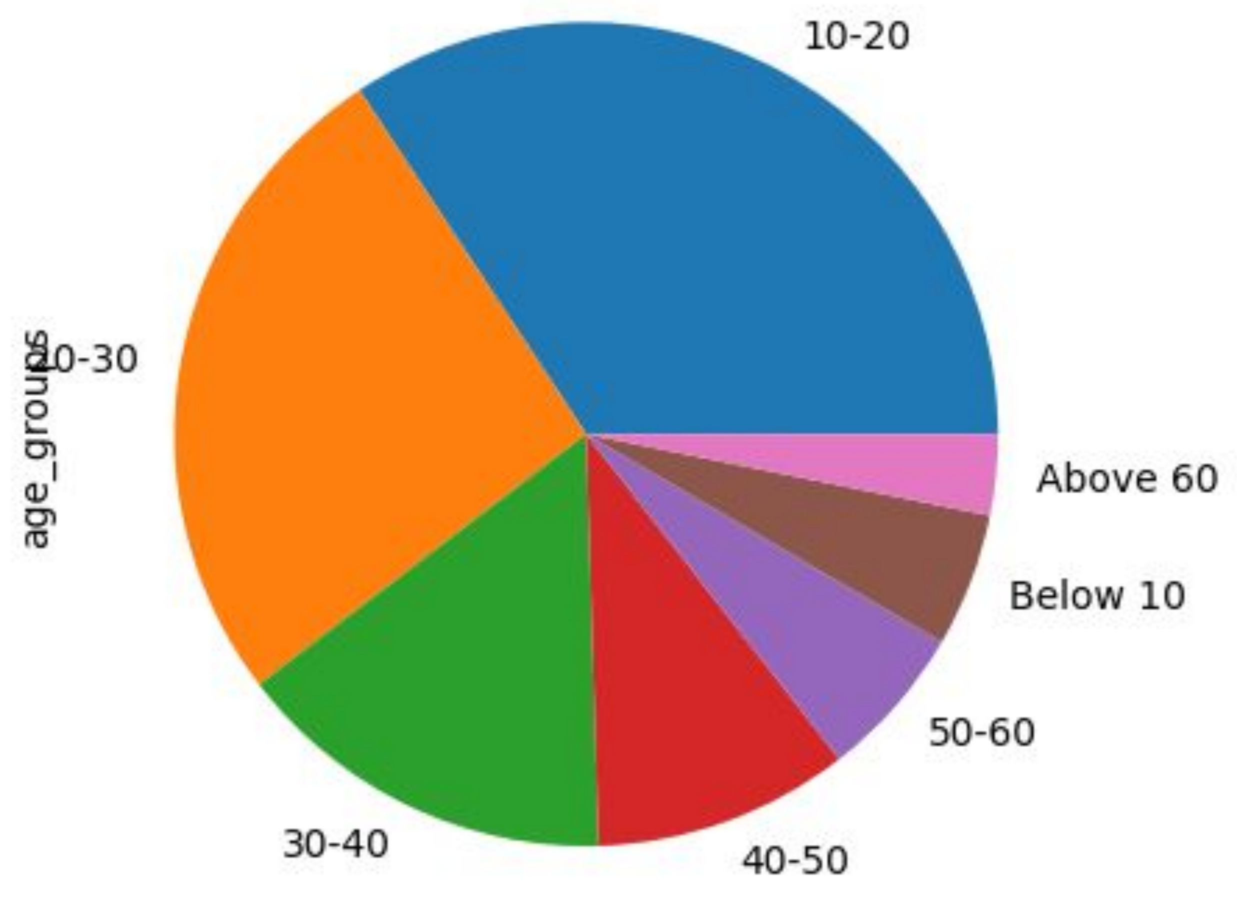
Value counts per year for the top 20



4. EXPLORATORY DATA ANALYSIS

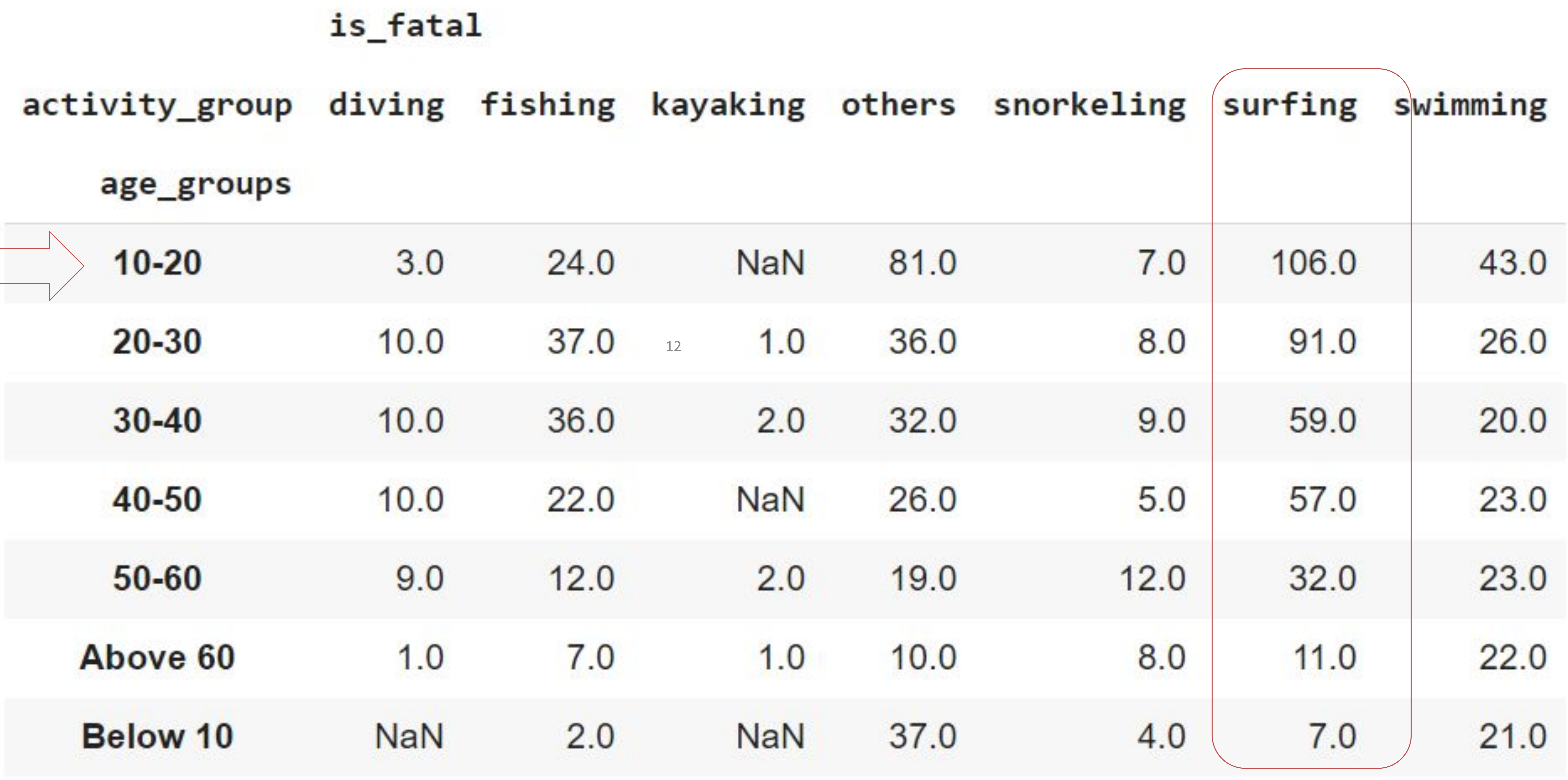


Number of cases / gender



Number of cases / age groups

5. Fatal cases per age and activity (last 10 years)



	is_fatal						
activity_group	diving	fishing	kayaking	others	snorkeling	surfing	swimming
age_groups							
10-20	3.0	24.0	NaN	81.0	7.0	106.0	43.0
20-30	10.0	37.0	1.0	36.0	8.0	91.0	26.0
30-40	10.0	36.0	2.0	32.0	9.0	59.0	20.0
40-50	10.0	22.0	NaN	26.0	5.0	57.0	23.0
50-60	9.0	12.0	2.0	19.0	12.0	32.0	23.0
Above 60	1.0	7.0	1.0	10.0	8.0	11.0	22.0
Below 10	NaN	2.0	NaN	37.0	4.0	7.0	21.0

5. Most risky profile

MOST RISKY PROFILE

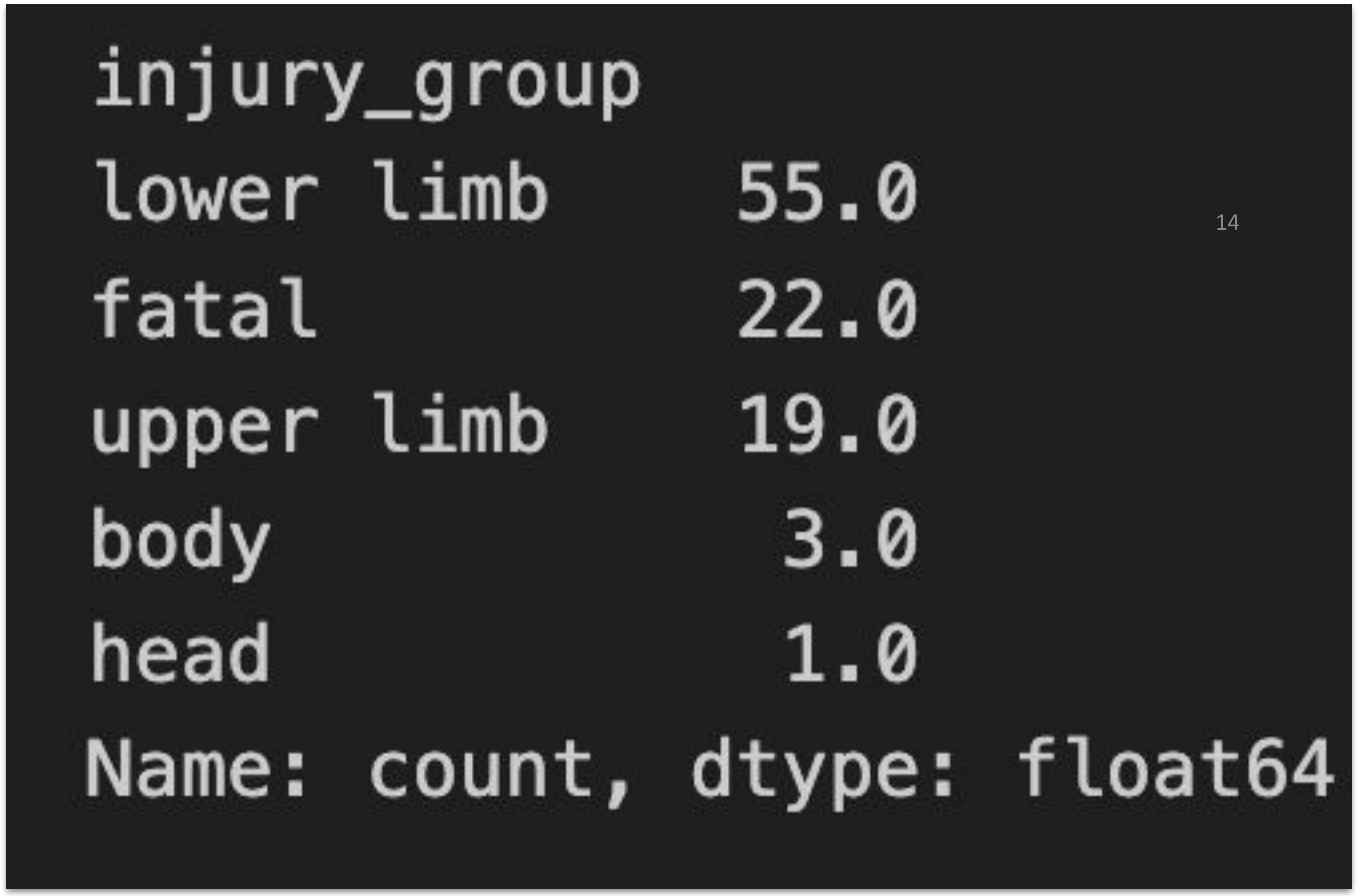
⇒ Male

⇒ Age groupe : 10-20 y. old

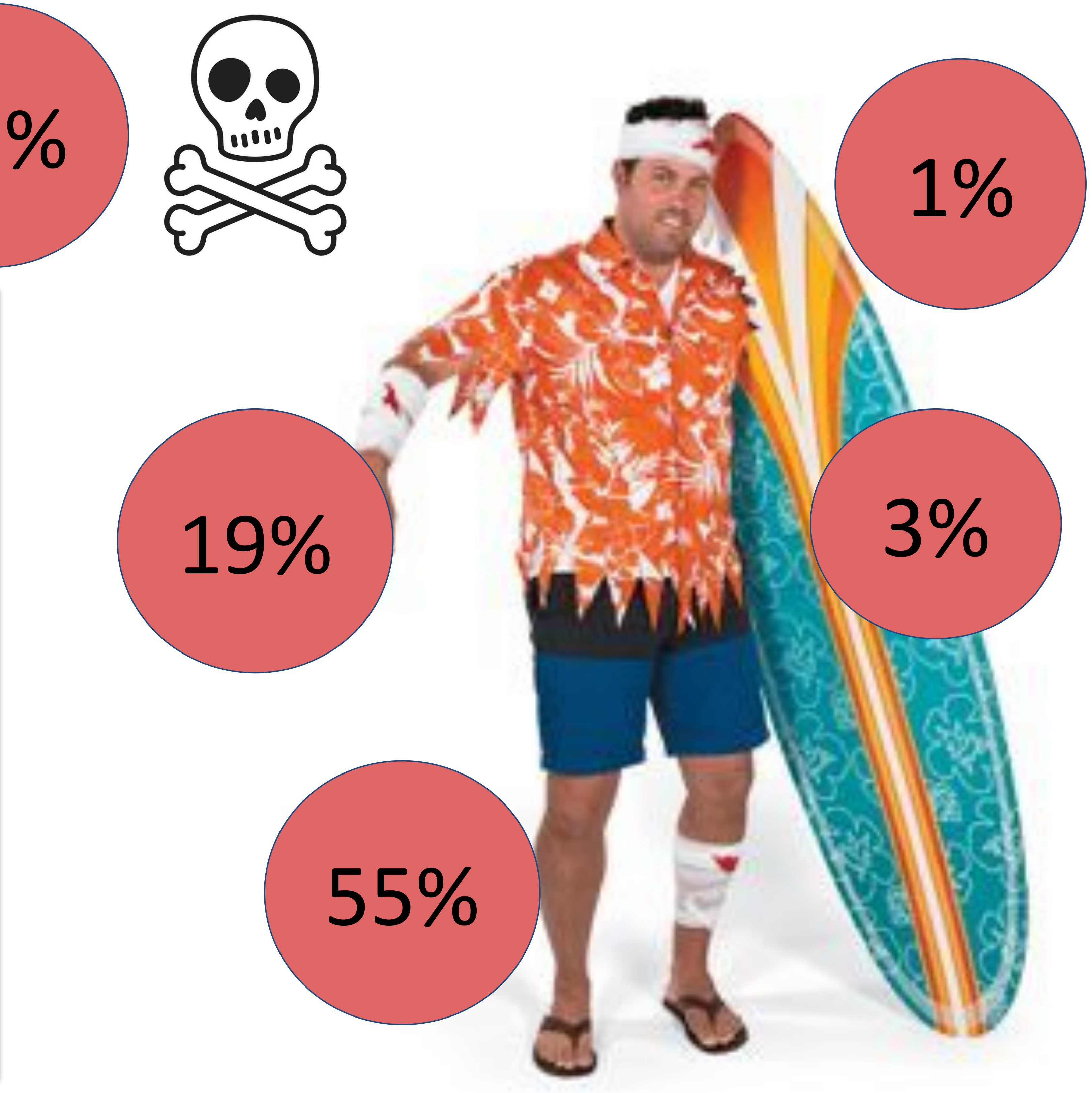
⇒ Surfing

⇒ in the USA (Florida)

Shark’s favorite body parts



```
injury_group
lower limb     55.0
fatal          22.0
upper limb     19.0
body            3.0
head            1.0
Name: count, dtype: float64
```



6. MAJOR CHALLENGES & NEXT STEPS

MAIN OBSTACLES

⇒ Distributing the timing

⇒ Having a robust hypothesis

⇒ We were confused about what we actually wanted to do = communication

NEXT STEPS

⇒ Create the different profiles based on risks

⇒ Create weighted average on the main risk factors to set prices for the insurance

THANK YOU FOR TRAVELLING WITH US !

7. CONCLUSION AND INSIGHTS

INSIGHTS

⇒ Most cases are men surfers in USA

⇒ Top 5 countries with most cases : USA, Australia, South Africa, New Zealand, Mexico

⇒ Age group is : Teenagers

2. PROJECT OVERVIEW

ORIGINAL DATASET

SHAPE ?

Description de l'activité 1

# OF MISSING DATAS

?

Description de l'activité 2

?????

Description de l'activité 3

OUR POSITION

TRAVEL INSURRANCE

CIE

Reveal patterns so we can adapt the insurance offers to clients based on the risk profil

HYPOTHESIS

Locations, gender and age are determining to understand risk profil

Relevant columns categories for analysis :

- Gender, Age, Country and State, Activity, Type of injury, Dates

DATA CLEANING

SUPPRESSION??

- Columns ?
- Rows ?

TECHNICS ???

Description de l'activité 2

METHODS

???

Description de l'activité 3

7. UMA : TRAVEL INSURANCE

	INSURANCE PACKAGE
RISK PROFIL ?????	600 €
VERY HIGH	1250 €
HIGH	650 €
MODERATE	300 €
LOW	500€