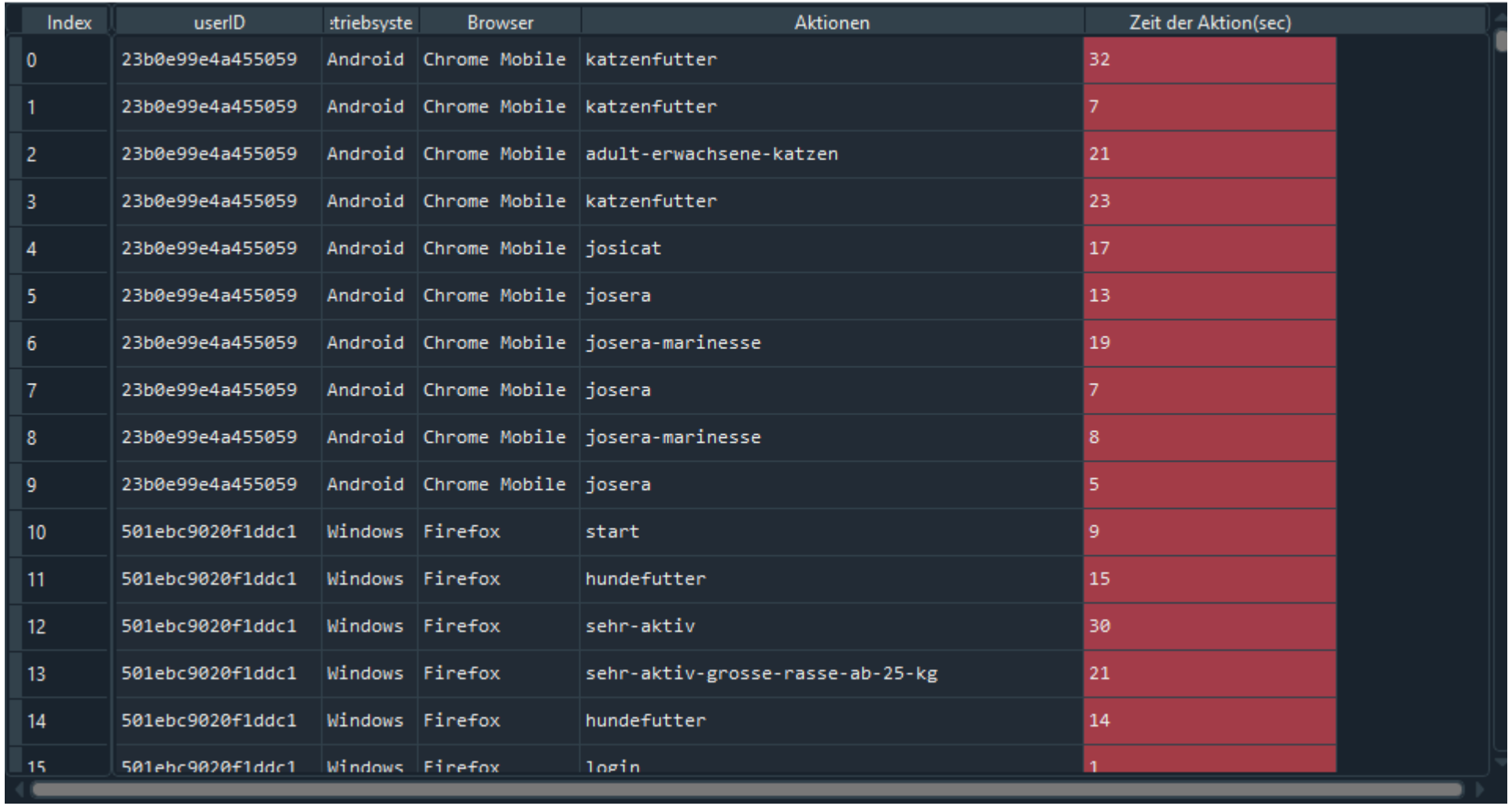
userID	Betriebsystem	Browser	Aktionen	Zeit der Aktion(sec)
23b0e99e4a455059	Android	Chrome Mobile	www.josera.de/katzenfutter.html	32
23b0e99e4a455059	Android	Chrome Mobile	www.josera.de/katzenfutter.html	7
23b0e99e4a455059	Android	Chrome Mobile	www.josera.de/katzenfutter/adult-erwachsene-katzen.html	21
23b0e99e4a455059	Android	Chrome Mobile	www.josera.de/katzenfutter.html	23
23b0e99e4a455059	Android	Chrome Mobile	www.josera.de/katzenfutter/josicat.html	17
23b0e99e4a455059	Android	Chrome Mobile	www.josera.de/katzenfutter/josera.html	13
23b0e99e4a455059	Android	Chrome Mobile	www.josera.de/josera-marinesse.html	19
23b0e99e4a455059	Android	Chrome Mobile	www.josera.de/katzenfutter/josera.html	7
23b0e99e4a455059	Android	Chrome Mobile	www.josera.de/josera-marinesse.html	8
23b0e99e4a455059	Android	Chrome Mobile	www.josera.de/katzenfutter/josera.html	5
501ebc9020f1ddc1	Windows	Firefox	www.josera.de/	9

1. Step

```
df = pd.read_csv('matomoDaten.csv', delimiter=';')

df_test = pd.read_csv('matomo_test.csv', delimiter=';')

df = df.dropna()



# select action from url
def filter_action(df):
    all_actions = list()
    for element in df['Aktionen']:

        #last element in the url
        action = element.split('/')[-1]
        precedent_action = element.split('/')[-2]

        if action == '' or precedent_action in ['account', 'login']:
            action = precedent_action
            if action == 'www.josera.de':
                action = 'start'
            all_actions.append(action)
            continue
        action = action[:-5] if action[-5:]=='.html' else  action
        if '.html' in action:
            action = action[:action.index('.html')]
        all_actions.append(action)
    return all_actions
```

df - DataFrame



Index	userID	:triebsyste	Browser	Aktionen	Zeit der Aktion(sec)
0	23b0e99e4a455059	Android	Chrome Mobile	katzenfutter	32
1	23b0e99e4a455059	Android	Chrome Mobile	katzenfutter	7
2	23b0e99e4a455059	Android	Chrome Mobile	adult-erwachsene-katzen	21
3	23b0e99e4a455059	Android	Chrome Mobile	katzenfutter	23
4	23b0e99e4a455059	Android	Chrome Mobile	josicat	17
5	23b0e99e4a455059	Android	Chrome Mobile	josera	13
6	23b0e99e4a455059	Android	Chrome Mobile	josera-marinesse	19
7	23b0e99e4a455059	Android	Chrome Mobile	josera	7
8	23b0e99e4a455059	Android	Chrome Mobile	josera-marinesse	8
9	23b0e99e4a455059	Android	Chrome Mobile	josera	5
10	501ebc9020f1ddc1	Windows	Firefox	start	9
11	501ebc9020f1ddc1	Windows	Firefox	hundefutter	15
12	501ebc9020f1ddc1	Windows	Firefox	sehr-aktiv	30
13	501ebc9020f1ddc1	Windows	Firefox	sehr-aktiv-grosse-rasse-ab-25-kg	21
14	501ebc9020f1ddc1	Windows	Firefox	hundefutter	14
15	501ebc9020f1ddc1	Windows	Firefox	login	1

2. Step

```python
def join_action(action):
    action = action.split('-')
    action = ''.join(action)
    action = action.split('=')
    action = ''.join(action)
    action = action.split('_')
    action = ''.join(action)
    action = action.split('.')
    action = ''.join(action)
    action = action.split('?')
    action = ''.join(action)
    action = action.split('&')
    return ''.join(action)


def rewrite_action(df):
    df['Aktionen'] = list(map(join_action, df['Aktionen']))
    return df
```

df - DataFrame

Index	userID	:triebsyste	Browser	Aktionen	:der Aktion
0	23b0e99e4a455059	Android	Chrome Mobile	katzenfutter	32
1	23b0e99e4a455059	Android	Chrome Mobile	katzenfutter	7
2	23b0e99e4a455059	Android	Chrome Mobile	adulterwachsenekatzen	21
3	23b0e99e4a455059	Android	Chrome Mobile	katzenfutter	23
4	23b0e99e4a455059	Android	Chrome Mobile	josicat	17
5	23b0e99e4a455059	Android	Chrome Mobile	josera	13
6	23b0e99e4a455059	Android	Chrome Mobile	joseramarinesse	19
7	23b0e99e4a455059	Android	Chrome Mobile	josera	7
8	23b0e99e4a455059	Android	Chrome Mobile	joseramarinesse	8
9	23b0e99e4a455059	Android	Chrome Mobile	josera	5
10	501ebc9020f1ddc1	Windows	Firefox	start	9
11	501ebc9020f1ddc1	Windows	Firefox	hundefutter	15
12	501ebc9020f1ddc1	Windows	Firefox	sehraktiv	30
13	501ebc9020f1ddc1	Windows	Firefox	sehraktivgrosserasseab25kg	21

3. Step

```
tokenizer = Tokenizer()
tokenizer.fit_on_texts(df['Aktionen'])

df['Aktionen'] = tokenizer.texts_to_sequences(df['Aktionen'])

ordEncoder_betriebsystem = OrdinalEncoder()
ordEncoder_browser = OrdinalEncoder()

df["Betriebsystem"] = ordEncoder_betriebsystem.fit_transform(df[["Betriebsystem"]])
df["Browser"] = ordEncoder_browser.fit_transform(df[["Browser"]])

# transform [3] to 3 for exemple
def remove_braket(list):
    elt = list[0]
    return elt

df['Aktionen'] = list(map(remove_braket, df['Aktionen']))


del df['userID']

df.to_csv(r'dataset.csv', index=False, header=True)
```

df - DataFrame

Index	triebsyste	Browser	Aktionen	Zeit der Aktion(sec)
0	0	1	23	32
1	0	1	23	7
2	0	1	74	21
3	0	1	23	23
4	0	1	75	17
5	0	1	32	13
6	0	1	46	19
7	0	1	32	7
8	0	1	46	8
9	0	1	32	5
10	3	2	6	9
11	3	2	1	15
12	3	2	33	30
13	3	2	76	21
14	3	2	1	14
15	3	2	3	1