```
In [1]: import pandas as pd
        import numpy as np
        import altair as alt

        # Handle large data sets without embedding them in the notebook
        alt.data_transformers.enable('data_server')
        # Include an image for each plot since Gradescope only supports displaying plots as images
        alt.renderers.enable('mimetype')
```

Out[1]: RendererRegistry.enable('mimetype')

```
In [2]: data = {'name.with.dots': [1,2,2,3,3,3,4,4,4,4]}
        df = pd.DataFrame(data)
        df
```
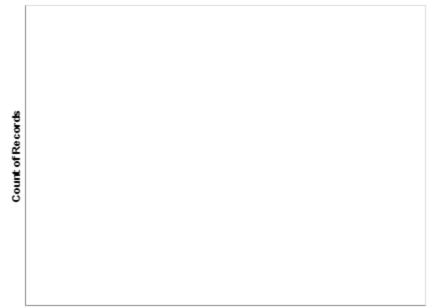
Out[2]:

	name.with.dots
0	1
1	2
2	2
3	3
4	3
5	3
6	4
7	4
8	4
9	4

```
In [3]: doesnt_work = alt.Chart(df).mark_bar().encode(
            x=alt.X('name.with.dots', bin=alt.Bin(maxbins=20)),
            y='count()'
        )

        doesnt_work
```

C:\Users\jakob\miniconda3\envs\D571\lib\site-packages\altair\utils\core.py:317: FutureWarning: i
teritems is deprecated and will be removed in a future version. Use .items instead.
  for col_name, dtype in df.dtypes.iteritems():

Count of Records

name.with.dots (binned)

In [4]:
```python
df = df.rename(columns={'name.with.dots':'name_with_underscores'})

does_work = alt.Chart(df).mark_bar().encode(
    x=alt.X('name_with_underscores', bin=alt.Bin(maxbins=20)),
    y='count()'
)

does_work
```

C:\Users\jakob\miniconda3\envs\D571\lib\site-packages\altair\utils\core.py:317: FutureWarning: i
teritems is deprecated and will be removed in a future version. Use .items instead.
  for col_name, dtype in df.dtypes.iteritems():

Out[4]:

name_with_underscores (binned)