```
import pandas as pd
import seaborn as sns
```

```
df=pd.read_csv("1651277648862_healthinsurance.csv")
```

```
df
```

	age	sex	weight	bmi	hereditary_diseases	no_of_dependents	smoker	city	bloodpressure	diabetes	regular_ex	
0	60.0	male	64	24.3	NoDisease	1	0	NewYork	72	0	0	
1	49.0	female	75	22.6	NoDisease	1	0	Boston	78	1	1	
2	32.0	female	64	17.8	Epilepsy	2	1	Phildelphia	88	1	1	
3	61.0	female	53	36.4	NoDisease	1	1	Pittsburg	72	1	0	
4	19.0	female	50	20.6	NoDisease	0	0	Buffalo	82	1	0	
...	...	...	...	...	...	...	...	...	...	...	...	
14995	39.0	male	49	28.3	NoDisease	1	1	Florence	54	1	0	
14996	39.0	male	74	29.6	NoDisease	4	0	Miami	64	1	0	
14997	20.0	male	62	33.3	NoDisease	0	0	Tampa	52	1	0	Fas
14998	52.0	male	88	36.7	NoDisease	0	0	PanamaCity	70	1	0	
14999	52.0	male	57	26.4	NoDisease	3	0	Kingsport	72	1	0	

15000 rows × 13 columns

Next steps: Generate code with df | New interactive sheet

```
df.head()
```

	age	sex	weight	bmi	hereditary_diseases	no_of_dependents	smoker	city	bloodpressure	diabetes	regular_ex	job_ti
0	60.0	male	64	24.3	NoDisease	1	0	NewYork	72	0	0	A
1	49.0	female	75	22.6	NoDisease	1	0	Boston	78	1	1	Engi
2	32.0	female	64	17.8	Epilepsy	2	1	Phildelphia	88	1	1	Academi
3	61.0	female	53	36.4	NoDisease	1	1	Pittsburg	72	1	0	(
4	19.0	female	50	20.6	NoDisease	0	0	Buffalo	82	1	0	HomeMa

Next steps: Generate code with df | New interactive sheet

```
df.tail()
```

	age	sex	weight	bmi	hereditary_diseases	no_of_dependents	smoker	city	bloodpressure	diabetes	regular_ex	
14995	39.0	male	49	28.3	NoDisease	1	1	Florence	54	1	0	
14996	39.0	male	74	29.6	NoDisease	4	0	Miami	64	1	0	
14997	20.0	male	62	33.3	NoDisease	0	0	Tampa	52	1	0	Fash
14998	52.0	male	88	36.7	NoDisease	0	0	PanamaCity	70	1	0	
14999	52.0	male	57	26.4	NoDisease	3	0	Kingsport	72	1	0	

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15000 entries, 0 to 14999
Data columns (total 13 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   age                  14604 non-null  float64
 1   sex                  15000 non-null  object
```

```
 2   weight               15000 non-null  int64
 3   bmi                  14044 non-null  float64
 4   hereditary_diseases  15000 non-null  object
 5   no_of_dependents     15000 non-null  int64
 6   smoker               15000 non-null  int64
 7   city                 15000 non-null  object
 8   bloodpressure        15000 non-null  int64
 9   diabetes             15000 non-null  int64
 10  regular_ex           15000 non-null  int64
 11  job_title            15000 non-null  object
 12  claim                15000 non-null  float64
dtypes: float64(3), int64(6), object(4)
memory usage: 1.5+ MB
```

```
df.describe()
```

	age	weight	bmi	no_of_dependents	smoker	bloodpressure	diabetes	regular_ex	cla
count	14604.000000	15000.000000	14044.000000	15000.000000	15000.000000	15000.000000	15000.000000	15000.000000	15000.0000
mean	39.547521	64.909600	30.266413	1.129733	0.198133	68.650133	0.777000	0.224133	13401.4376
std	14.015966	13.701935	6.122950	1.228469	0.398606	19.418515	0.416272	0.417024	12148.2396
min	18.000000	34.000000	16.000000	0.000000	0.000000	0.000000	0.000000	0.000000	1121.9000
25%	27.000000	54.000000	25.700000	0.000000	0.000000	64.000000	1.000000	0.000000	4846.9000
50%	40.000000	63.000000	29.400000	1.000000	0.000000	71.000000	1.000000	0.000000	9545.6500
75%	52.000000	76.000000	34.400000	2.000000	0.000000	80.000000	1.000000	0.000000	16519.1250
max	64.000000	95.000000	53.100000	5.000000	1.000000	122.000000	1.000000	1.000000	63770.4000

```
df.shape
```

```
(15000, 13)
```

```
df.size
```

```
195000
```

```
df["age"].mean()
```

```
np.float64(39.54752122706108)
```

```
df.isnull().sum()
```

	0
age	396
sex	0
weight	0
bmi	956
hereditary_diseases	0
no_of_dependents	0
smoker	0
city	0
bloodpressure	0
diabetes	0
regular_ex	0
job_title	0
claim	0

dtype: int64

```
df.nunique()
```

	0
age	47
sex	2
weight	58
bmi	269
hereditary_diseases	10
no_of_dependents	6
smoker	2
city	91
bloodpressure	69
diabetes	2
regular_ex	2
job_title	35
claim	2054

dtype: int64

```
df["age"].value_counts()
```

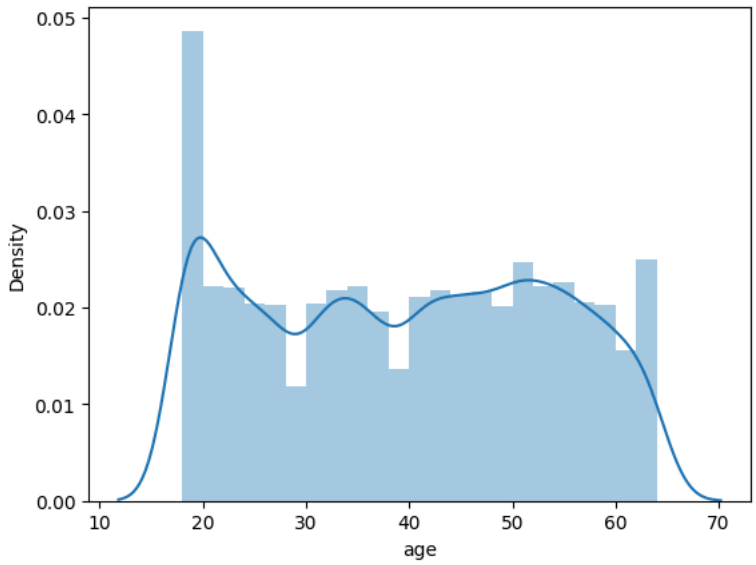
age	count
18.0	768
19.0	652
50.0	363
51.0	357
54.0	348
56.0	341
42.0	340
52.0	339
22.0	333
53.0	309
30.0	306

```
sns.distplot(df["age"])
```

```
/tmp/ipython-input-316555093.py:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751
```

```
sns.distplot(df["age"])
```

```
<Axes: xlabel='age', ylabel='Density'>
```

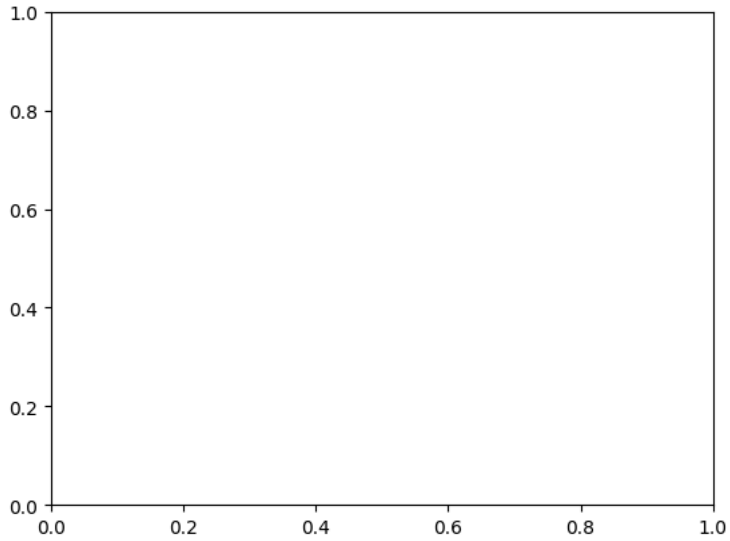



```
sns.distplot(df["sex"])
```

46.0	303
41.0	303
43.0	298
44.0	297
36.0	294
31.0	289
49.0	285
20.0	279
37.0	278
26.0	278
25.0	272
58.0	271
38.0	265
57.0	261
64.0	253
62.0	249
60.0	244
63.0	228
61.0	211

```
39.0    135
29.0     23
Name: count, dtype: int64
```

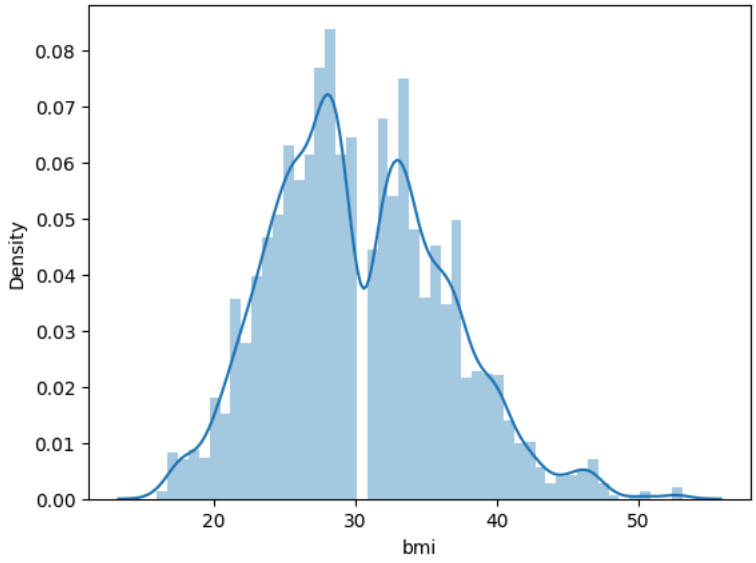
```
/tmp/ipython-input-4018499019.py:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df["sex"])
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
/tmp/ipython-input-4018499019.py in <cell line: 0>()
----> 1 sns.distplot(df["sex"])

                                 1 frames
/usr/local/lib/python3.12/dist-packages/pandas/core/series.py in __array__(self, dtype, copy)
   1029         """
   1030         values = self._values
-> 1031         arr = np.asarray(values, dtype=dtype)
   1032         if using_copy_on_write() and astype_is_view(values.dtype, arr.dtype):
   1033             arr = arr.view()

ValueError: could not convert string to float: 'male'
```



Next steps: Explain error

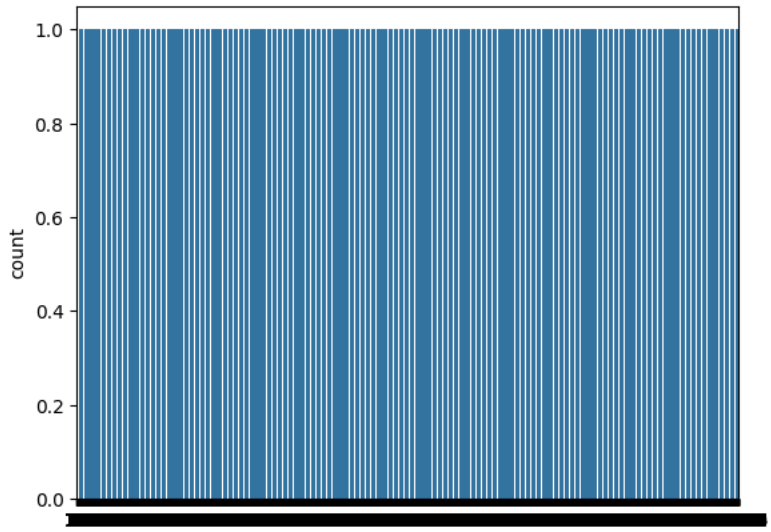
```
sns.distplot(df['bmi'])
```

```
/tmp/ipython-input-4168411822.py:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df['bmi'])
<Axes: xlabel='bmi', ylabel='Density'>
```



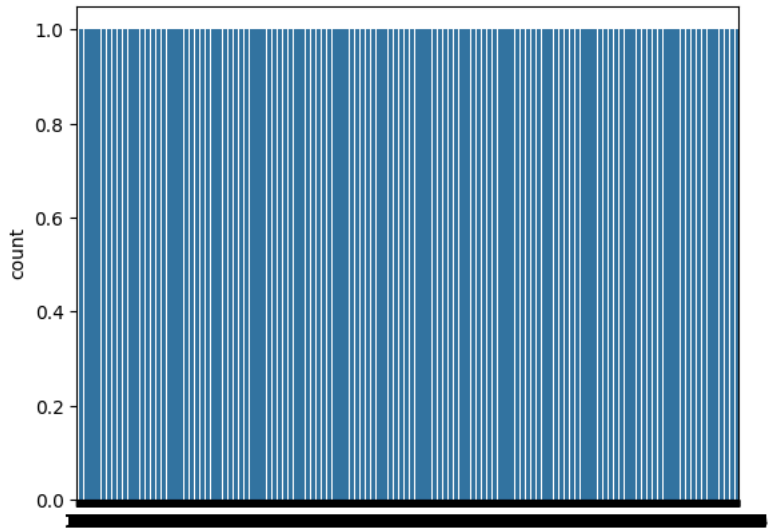

```
sns.countplot(df["age"])
```

```
<Axes: ylabel='count'>
```



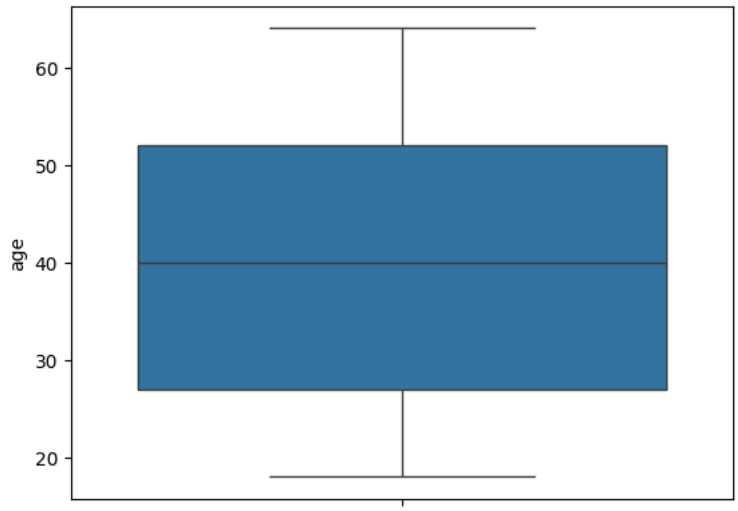

```
sns.countplot(df["bmi"])
```

```
<Axes: ylabel='count'>
```



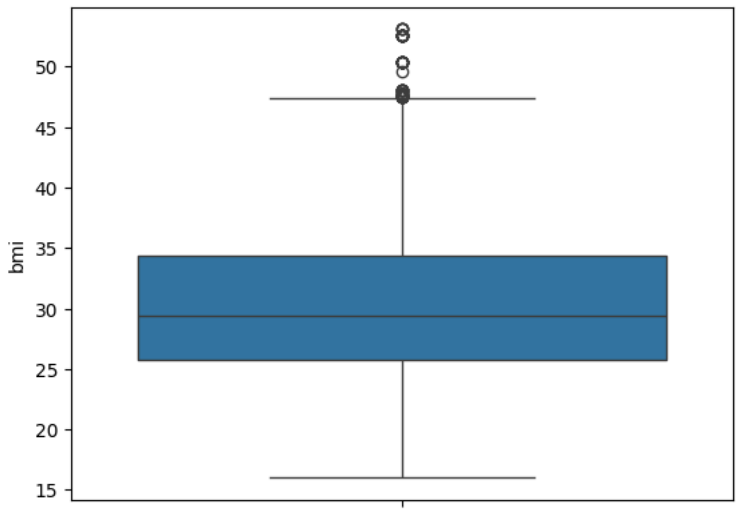

```
sns.boxplot(df["age"])
```

```
<Axes: ylabel='age'>
```




```
sns.boxplot(df["bmi"])
```

```
<Axes: ylabel='bmi'>
```




```
sns.boxplot(df["sex"])
```

```
<Axes: ylabel='sex'>
```