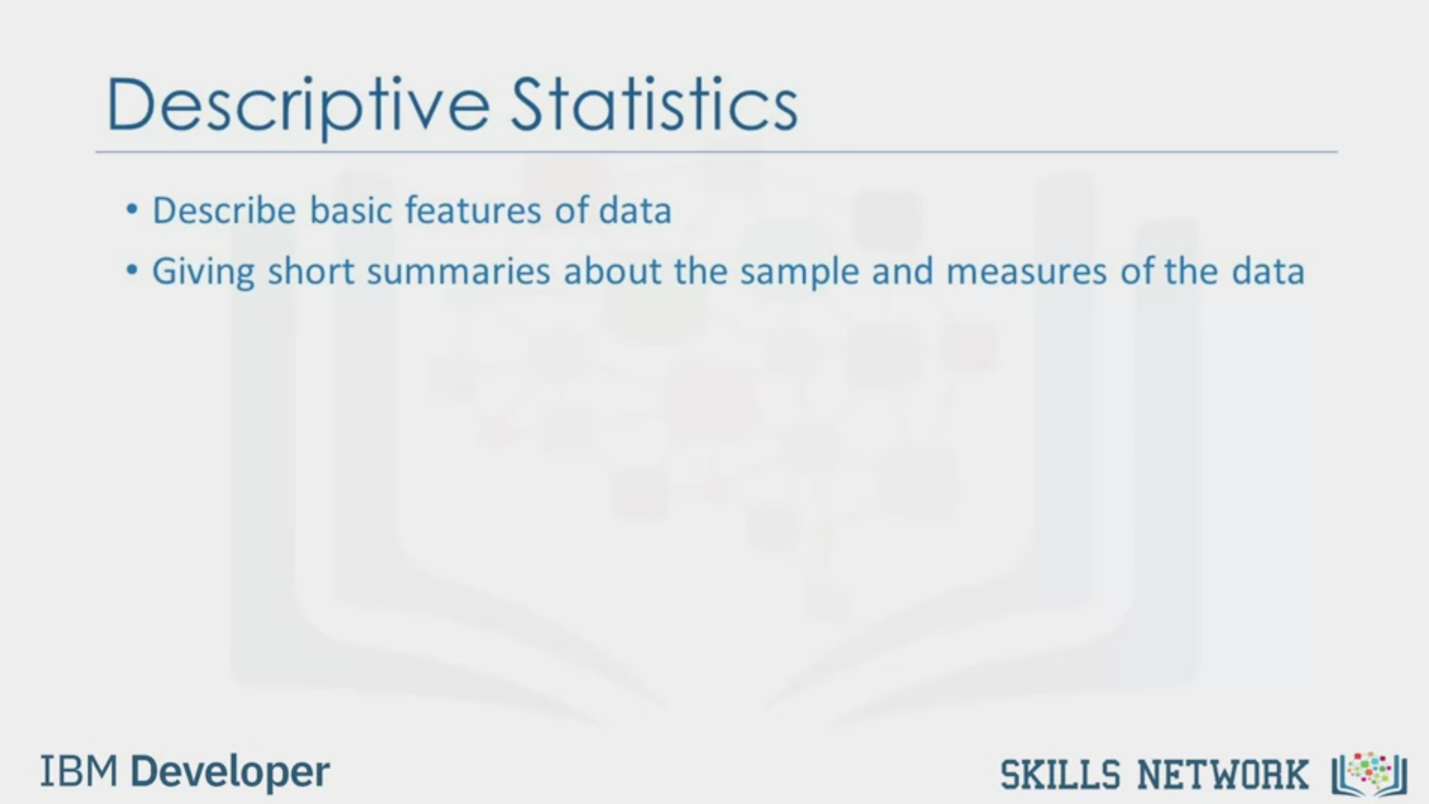


Descriptive Statistics

IBM Developer

SKILLS NETWORK



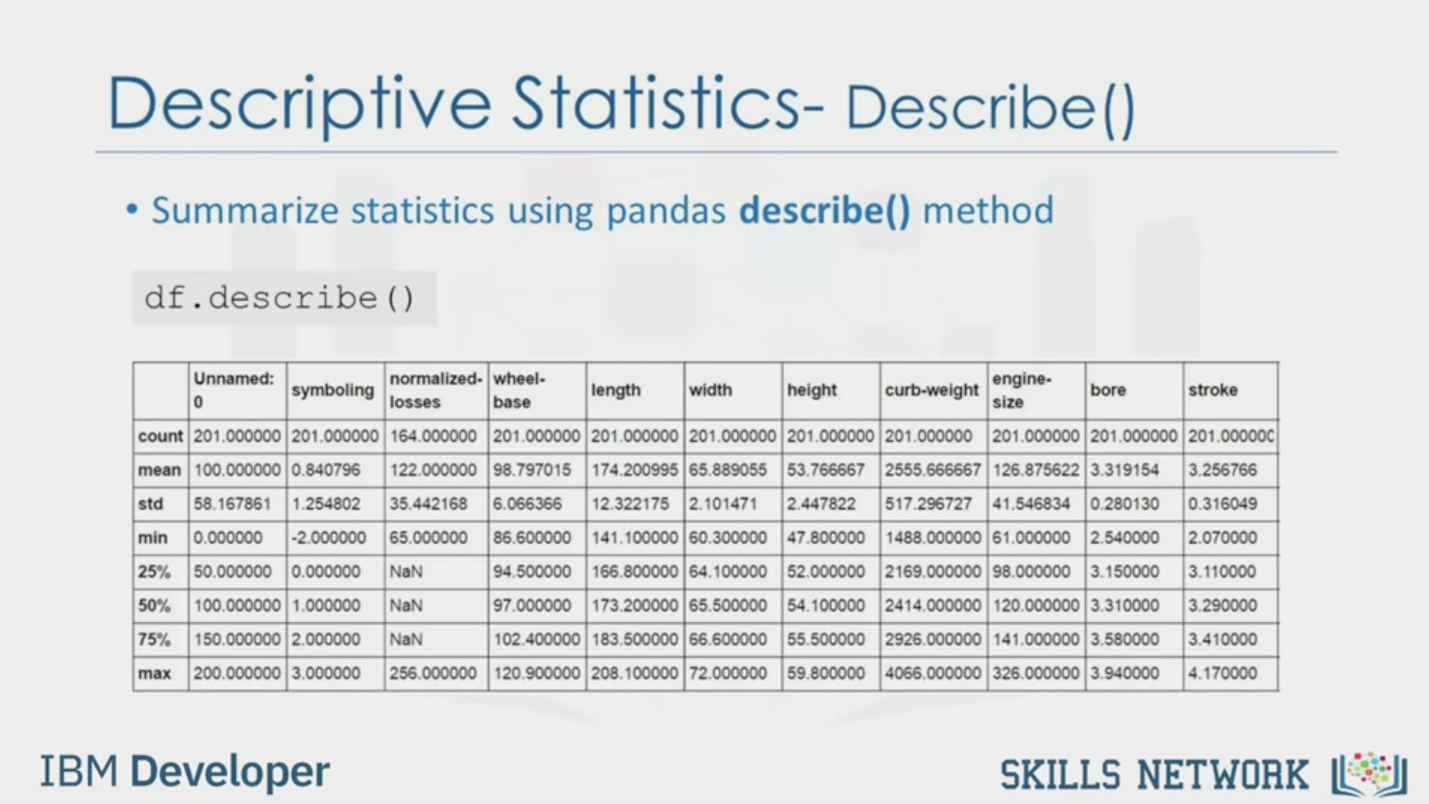
Descriptive Statistics

• Describe basic features of data

• Giving short summaries about the sample and measures of the data

IBM Developer

SKILLS NETWORK



Descriptive Statistics- Describel)

• Summarize statistics using pandas describe() method

df.describe ()

Unnamed:

normalized- wheel-

symboling

losses

1 base

length

width

height

engine•

curb-weight

size

bore

stroke

count 201.000000 201.000000 164.000000 201.000000 201.000000 201.000000 201.000000 201.000000 201.000000 201.000000 201.00000C

(mean 100.000000 0.840796

|122.000000 98.797015 174.200995 65.889055 53.766667 2555.666667 126.875622 3.319154

3.256766

Istd

58.167861 1.254802

35.442168

6.066366

12.322175 2.101471

2.447822

517.296727 41.546834 0.280130 0.316049

I min

0.000000

-2.000000 65.000000

86.600000 141.100000 60.300000 47.800000 1488.000000 61.000000 2.540000 2.070000

25%

50.000000 0.000000

I NaN

94.500000 166.800000 64.100000 52.000000 2169.000000 98.000000 3.150000

3.110000

50%

/100.000000 1.000000

NaN

(97.000000 173.200000 65.500000 | 54.100000 2414.000000 120.000000 3.310000

3.290000

75%

150.0000002.000000

I NaN

/102.400000 183.500000 66.600000 55.500000 2926.000000 141.000000 3.580000 3.410000

max

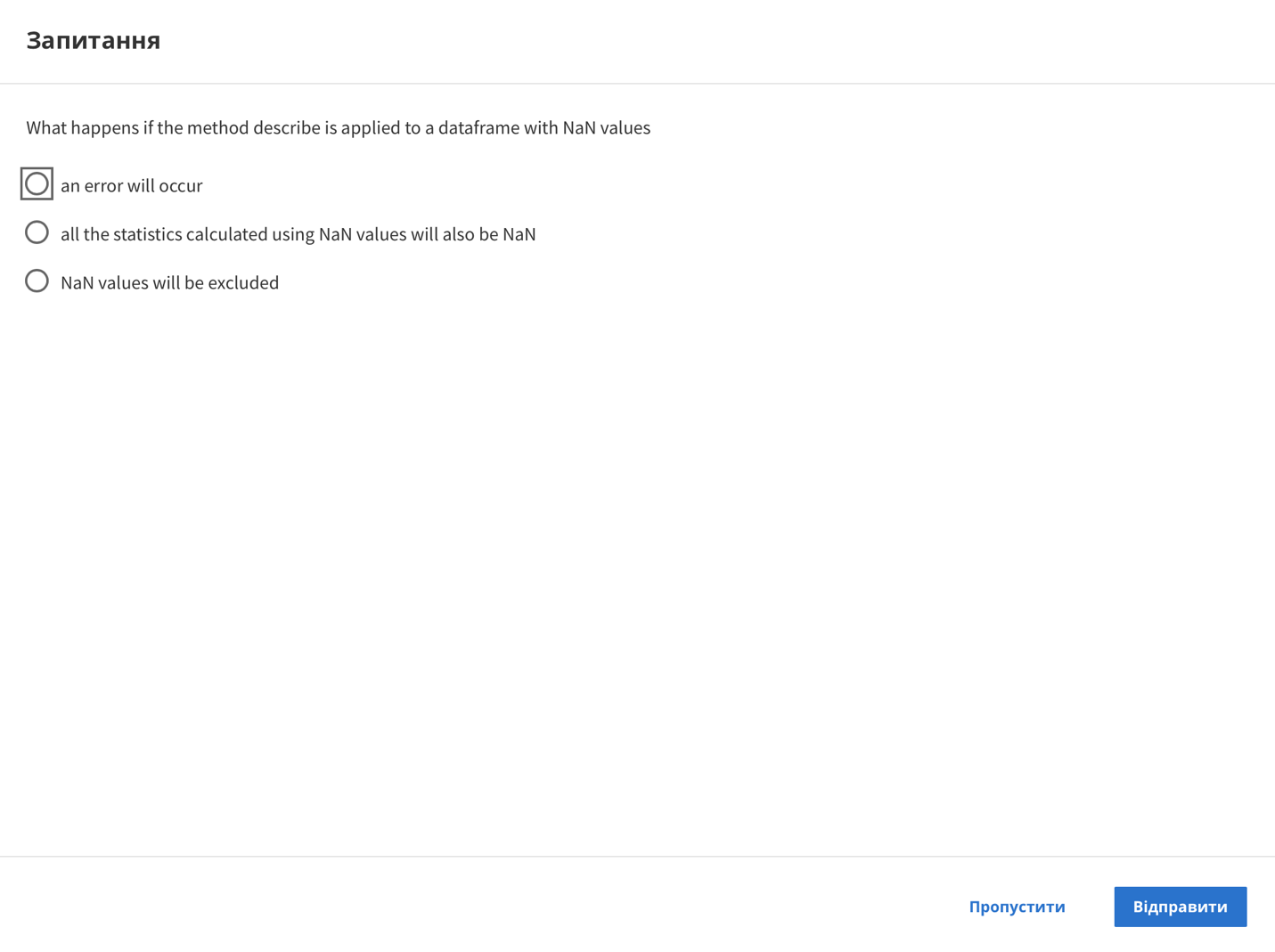
200.000000 3.000000

256.000000| 120.900000 208.100000 72.000000 59.800000 4066.000000 326.000000 3.940000

4.170000

IBM Developer

SKILLS NETWORK



Question

What happens if the method describe is applied to a dataframe with NaN values

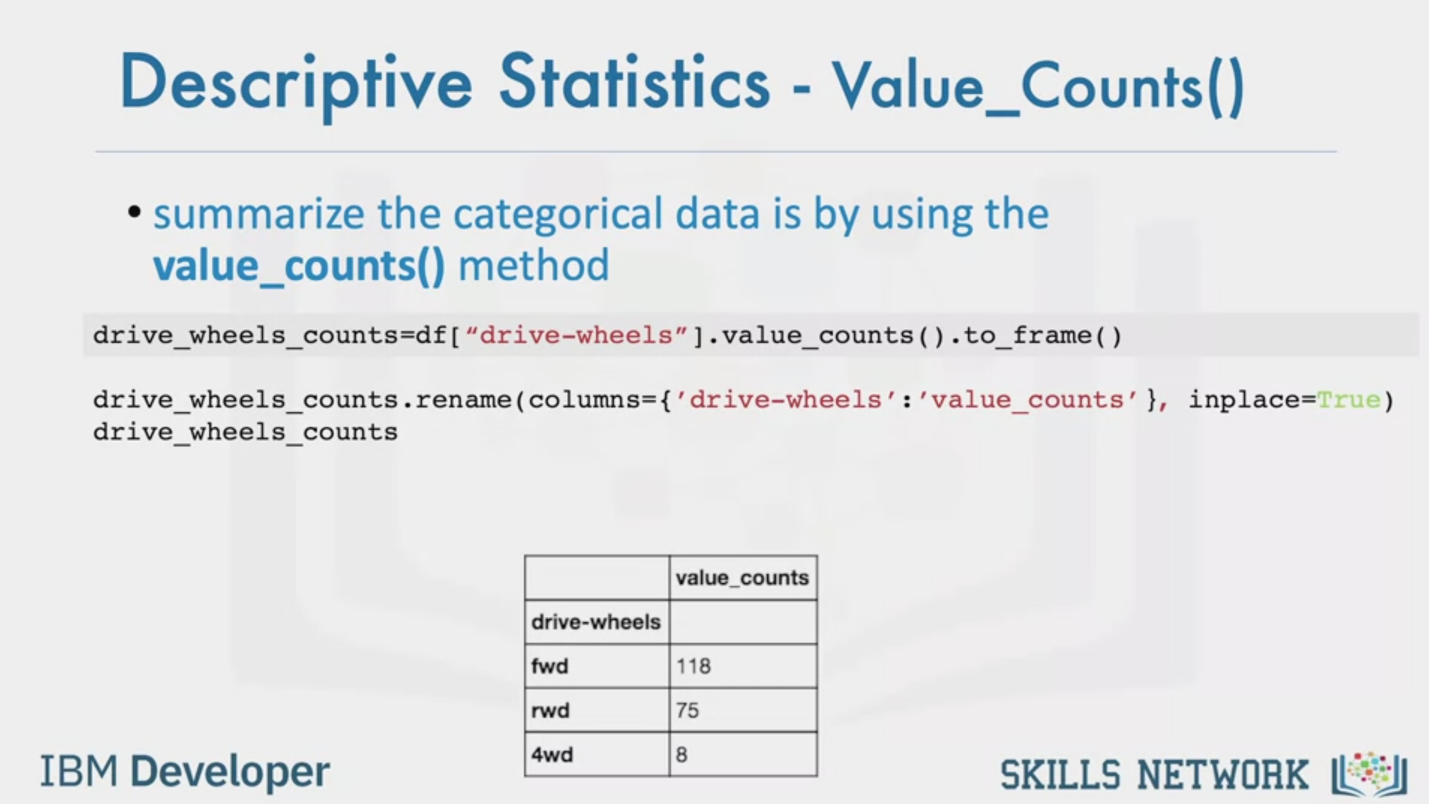
an error will occur

all the statistics calculated using NaN values will also be NaN

NaN values will be excluded

Skip

Send



Descriptive Statistics - Value\_Counts()

• summarize the categorical data is by using the

value\_counts() method

drive\_wheels\_counts=df["drive-wheels"].value\_counts ).to\_frame ()

drive\_wheels\_counts.rename (columns={'drive-wheels': 'value\_counts'}, inplace=True)

drive\_wheels\_counts

value\_counts

drive-wheels

fwd

rwd

4wd

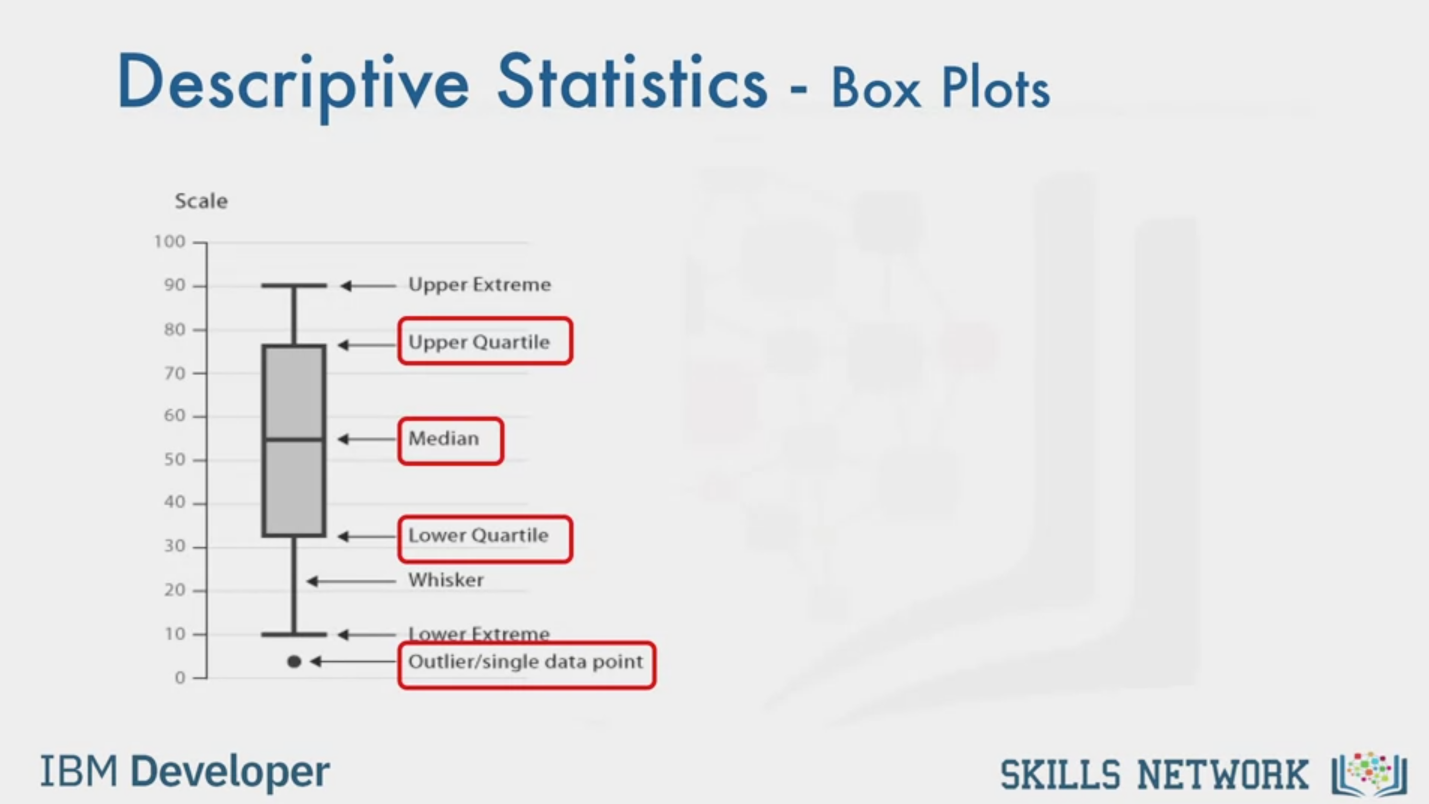
118

75

8

IBM Developer

SKILLS NETWORK



Descriptive Statistics - Box Plots

Scale

Upper Extreme

Upper Quartile

Median

Lower Quartile

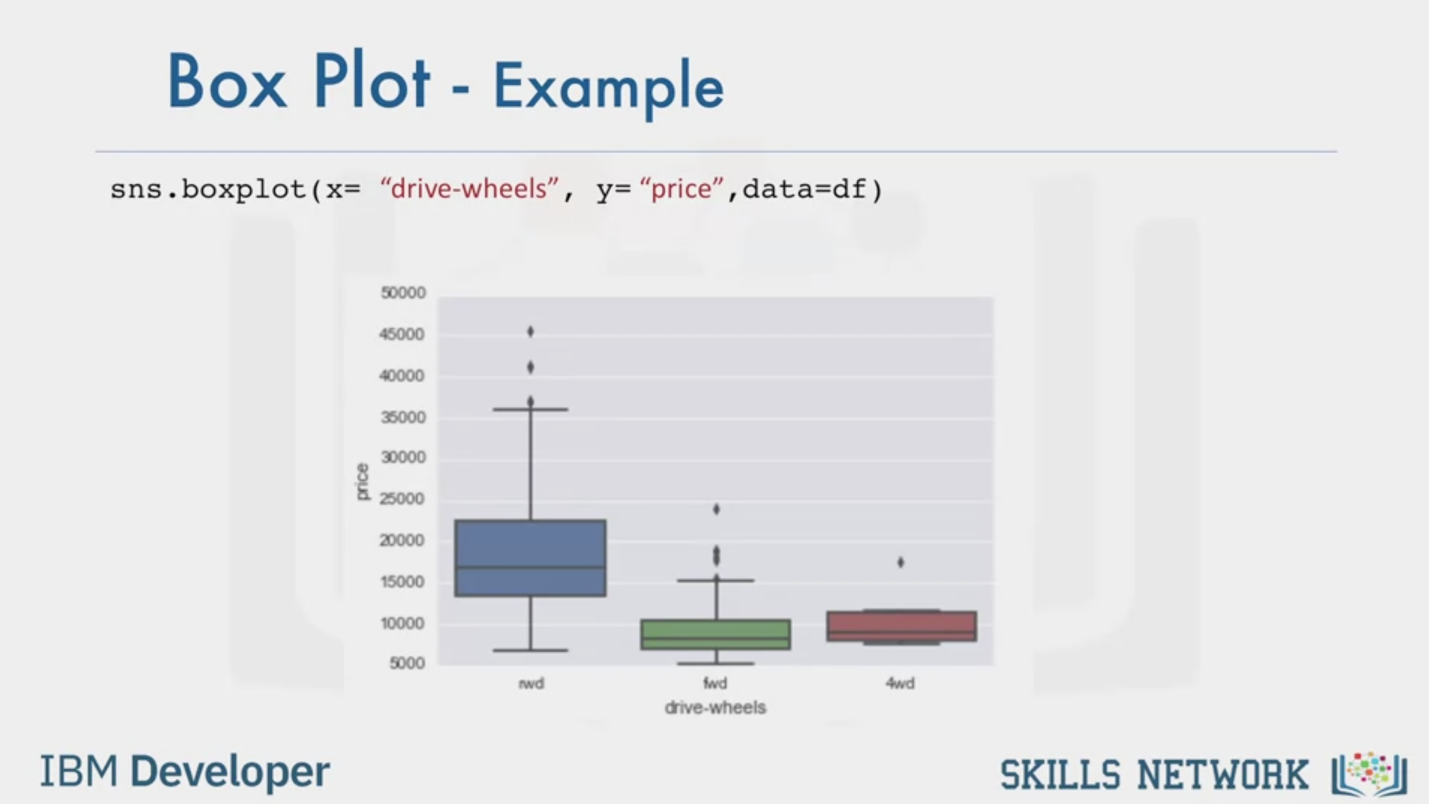
Whisker

Lower Extreme

Outlier/single data point

IBM Developer

SKILLS NETWORK



Box Plot - Example

sns.boxplot(x= "drive-wheels", y= "price", data=df)

rwd

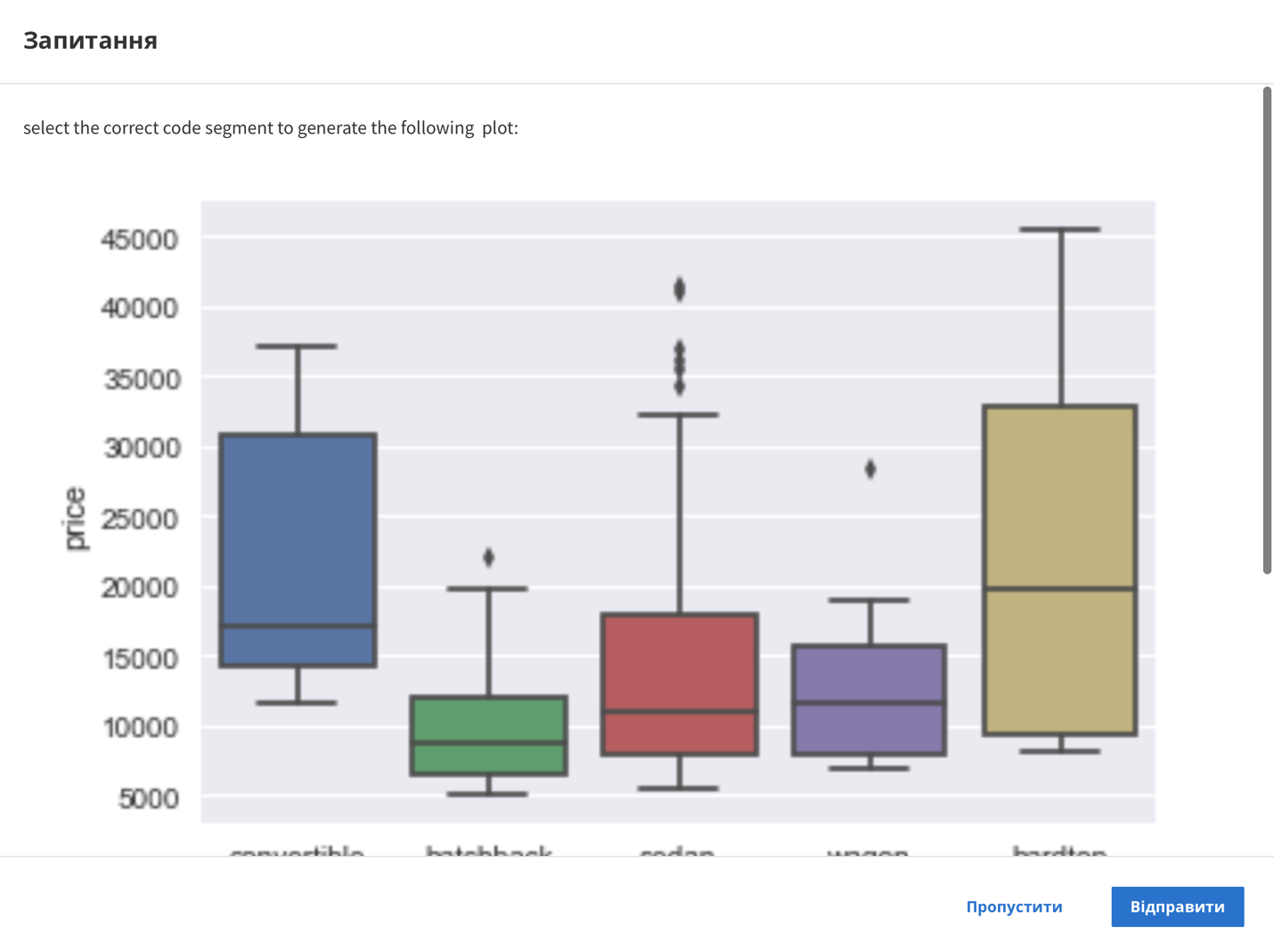
fwd

4wd

drive-wheels

IBM Developer

SKILLS NETWORK



Question

select the correct code segment to generate the following plot:



convertible

hatchback

sedan

wagon

hardtop

body-style

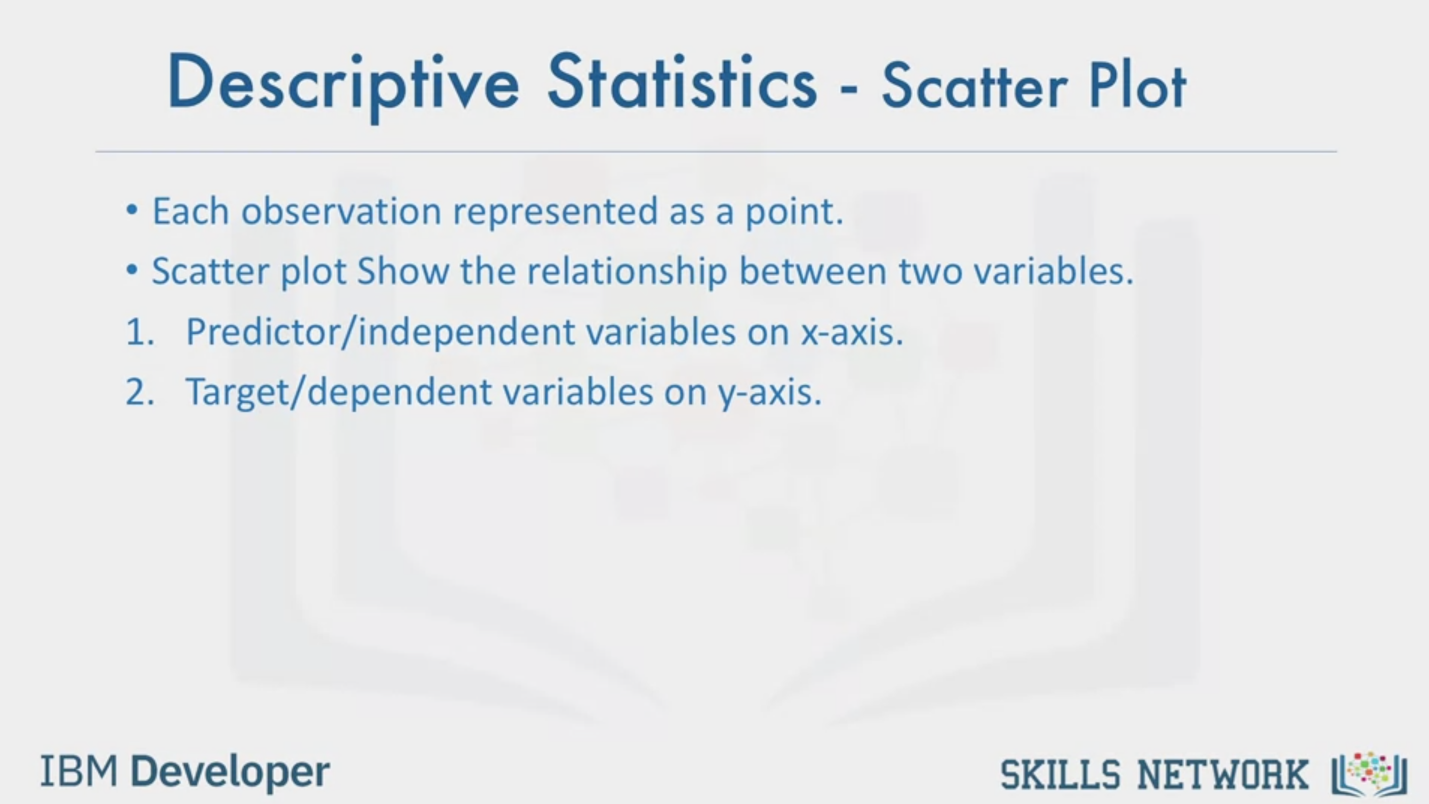
1 sns.boxplot (x="bodv-stvle", y="price", data=df)

1 sns.boxplot (x="engine-location", y="price", data=df)

1 sns.boxplot (x="drive-wheels", y="price", data=df)

Skip

Send



Descriptive Statistics - Scatter Plot

• Each observation represented as a point.

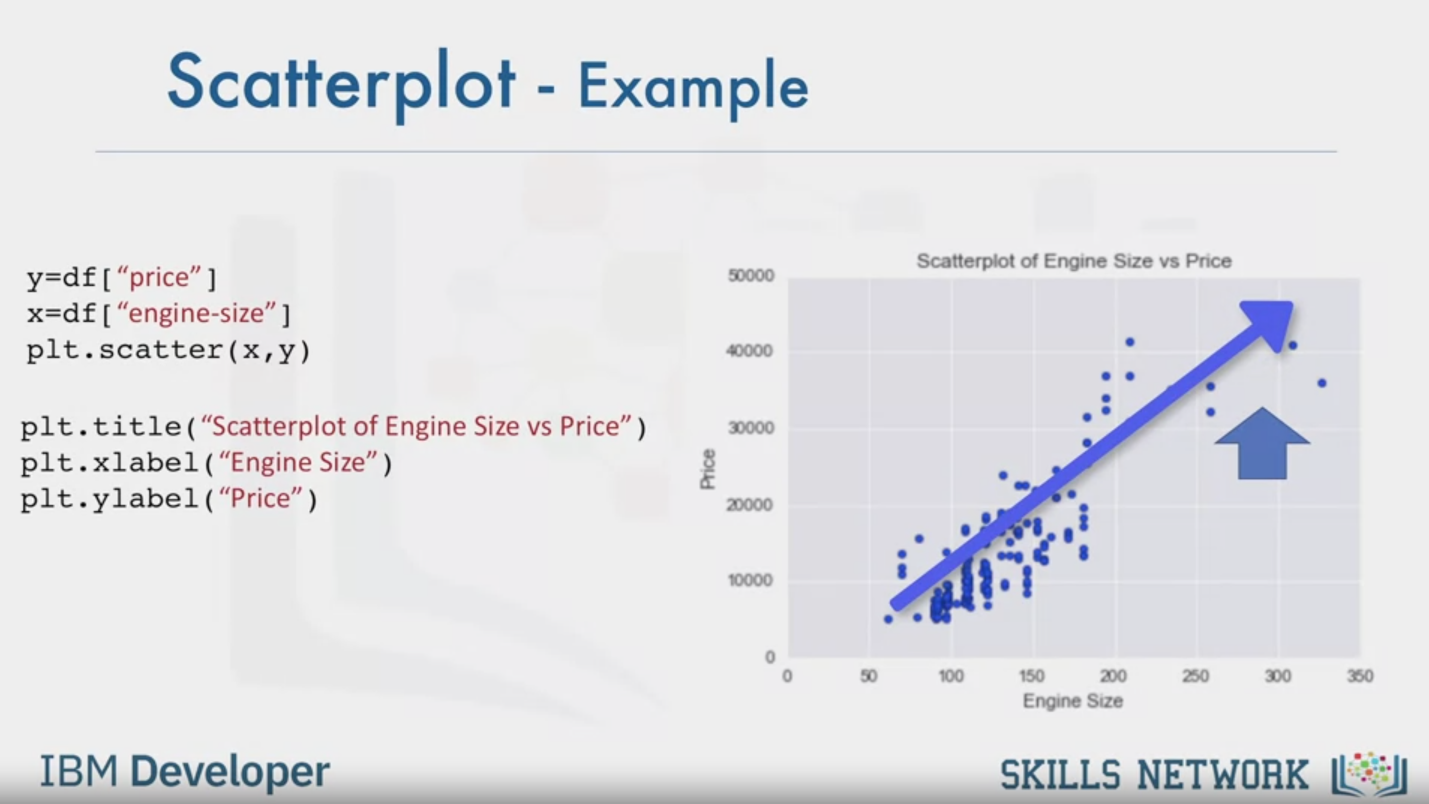
• Scatter plot Show the relationship between two variables.

1. Predictor/independent variables on x-axis.

2. Target/dependent variables on y-axis.

IBM Developer

SKILLS NETWORK



Scatterplot – Example

y=df ["price"]

x=df ["engine-size"]

plt.scatter (x, y)

plt.title ( "Scatterplot of Engine Size vs Price" )

plt.xlabel ("Engine Size" )

plt.ylabel ( "Price" )

Scatterplot of Engine Size vs Price

IBM Developer

SKILLS NETWORK



SKILLS NETWORK

IBM Developer

In this video, we'll be talking about Descriptive Statistics.

When you begin to analyze data, it's important to first explore your data

before you spend time building complicated models.

One easy way to do so, is to calculate some Descriptive Statistics for your data.

Descriptive statistical analysis helps to describe basic features of a data set, and

obtains a short summary about the sample and measures of the data.

Let's show you a couple different useful methods.

One way in which we can do this is by using the describe function in pandas.

Using the describe function and applying it on your data frame, the describe

function automatically computes basic statistics for all numerical variables.

It shows the mean, the total number of data points, the standard deviation,

the quartiles and the extreme values.

Any NAN values are automatically skipped in these statistics.

This function will give you a clear idea of the distribution of your different

variables.

You could have also categorical variables in your data set.

These are variables that can be divided up into different categories or groups, and

have discrete values.

For example, in our data set we have the drive system as a categorical variable,

which consists of the categories, forward wheel drive, rear wheel drive and

four wheel drive.

One way you can summarize the categorical data,

is by using the function value\_counts.

We can change the name of the column to make it easier to read.

We see that we have 118 cars in the front wheel drive category.

75 cars in the rear wheel drive category, and

8 cars in the four wheel drive category.

Box plots are a great way to visualize numeric data,

since you can visualize the various distributions of the data.

The main features that the box plot shows, are the median of the data,

which represents where the middle data point is.

The upper quartile shows where the 75th percentile is.

The lower quartile shows where the 25th percentile is.

The data between the upper and

lower quartile represents the interquartile range.

Next you have the lower and upper extremes.

These are calculated as 1.5 times the interquartile range,

above the 75th percentile, and as 1.5 times the IQR below the 25th percentile.

Finally, box plots also display outliers as individual dots that

occur outside the upper and lower extremes.

With box plots, you can easily spot outliers, and

also see the distribution and skewness of the data.

Box plots make it easy to compare between groups.

In this example, using box plot we can see the distribution of different categories

of the drive wheels feature over price feature.

We can see that the distribution of price between the rear wheel drive, and

the other categories are distinct.

But the price for front wheel drive and

four wheel drive are almost indistinguishable.

Often times we tend to see continuous variables in our data.

These data points are numbers contained in some range.

For example, in our data set price and engine size are continuous variables.

What if we want to understand the relationship between engine size and

price.

Could engine size possibly predict the price of a car?

One good way to visualize this is using a scatter plot.

Each observation in the scatter plot is represented as a point.

This plot shows the relationship between two variables.

The predictor variable, is the variable that you are using to predict an outcome.

In this case our predictor variable is the engine size.

The target variable is the variable that you are trying to predict.

In this case, our target variable is the price.

Since this would be the outcome.

In a scatter plot, we typically set the predictor variable on the x-axis or

horizontal axis, and we set the target variable on the y-axis or vertical axis.

In this case, we will thus plot the engine size on the x-axis and

the price on the y-axis.

We are using, the matplotlib functions scatter here, taking in x and y variable.

Something to note is that it's always important to label your axes, and

write a general plot title, so that you know what you're looking at.

Now how is the variable engine size related to price?

From the scatter plot, we see that as the engine size goes up,

the price of the car also goes up.

This is giving us an initial indication that there is a positive

linear relationship between these two variables.