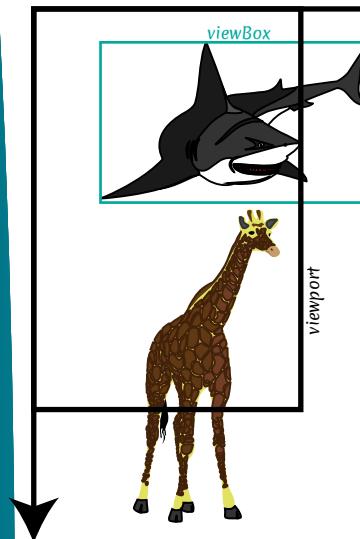


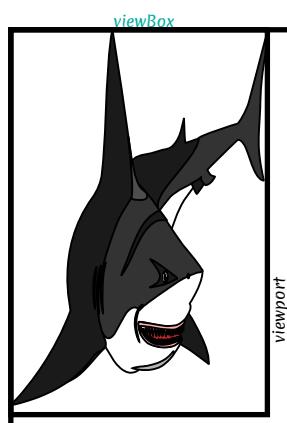
lesson 4

PRESERVEASPECTRATIO



In the previous lesson, the `viewBox` and the `viewport` have the same aspect ratio so we could reduce its size easily. What if they don't have the same aspect ratio? We could see the `svg` element that `viewport` is positioned vertically and its `viewBox` position horizontally. So their aspect ratios are different. We could see the shark that precisely fitted inside the `viewBox`, but we could see that giraffe head is also visible despite it located outside the `viewBox`. But why does it happen? This is a hidden behavior of the `preserveAspectRatio` attribute.

This attribute could take two parameters `<align>` and `<meetOrSlice>`. The first parameter let us chose the origin points of the `viewport` and `viewBox` that the user agent will use in the second step. This parameter could take 10 values.

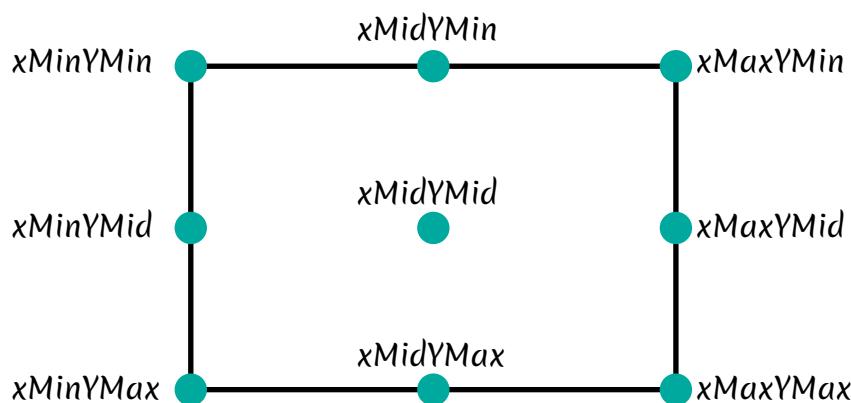


The value of “**none**” means that the aspect ratio will not preserve. The `viewBox` will fit inside the `viewport` with distortion of the graphics. if the `none` values is set any value of the second parameter will be ignored by user agent

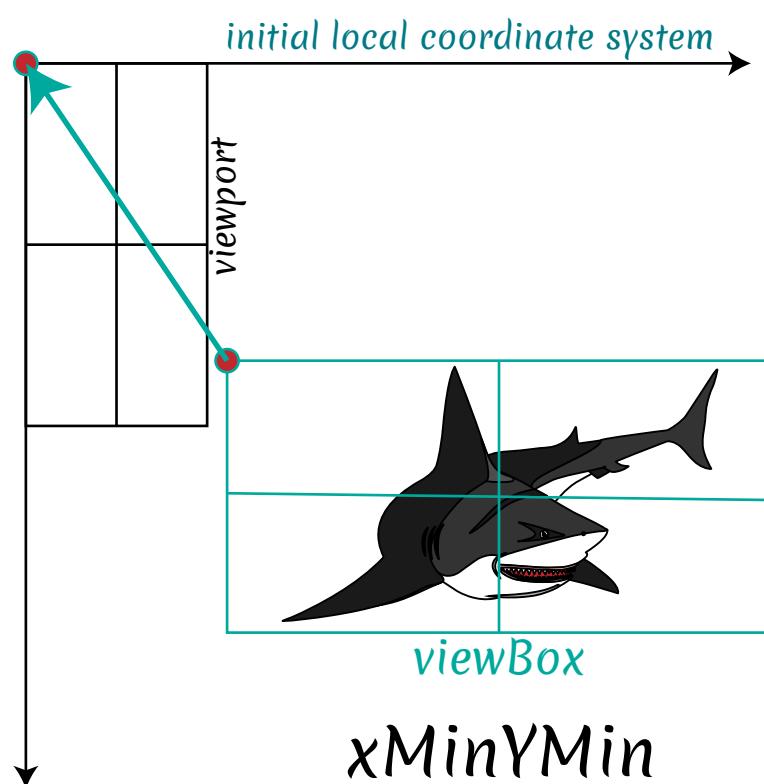
`preserveAspectRatio="none"`

lesson 4

PRESERVEASPECTRATIO

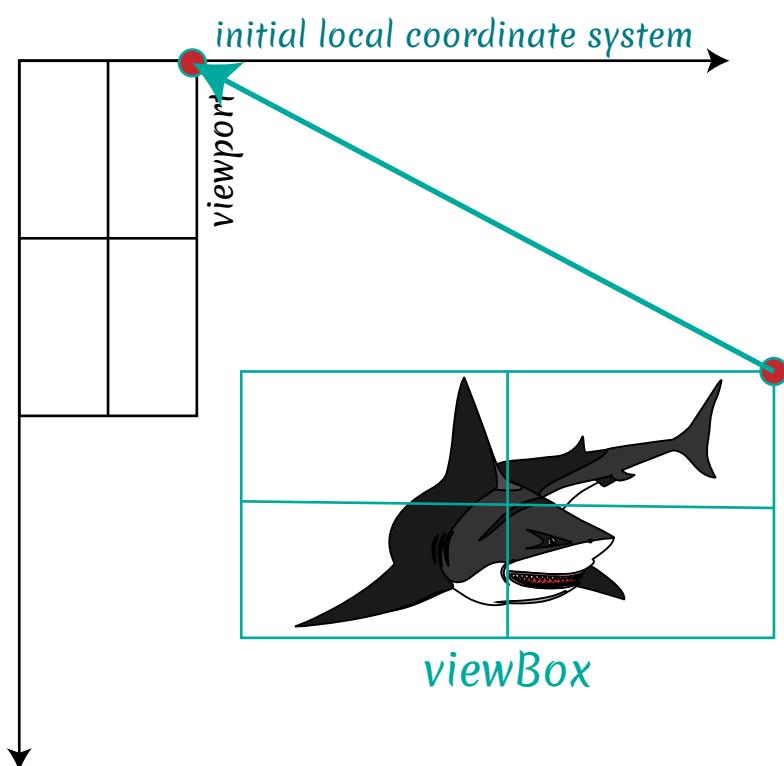
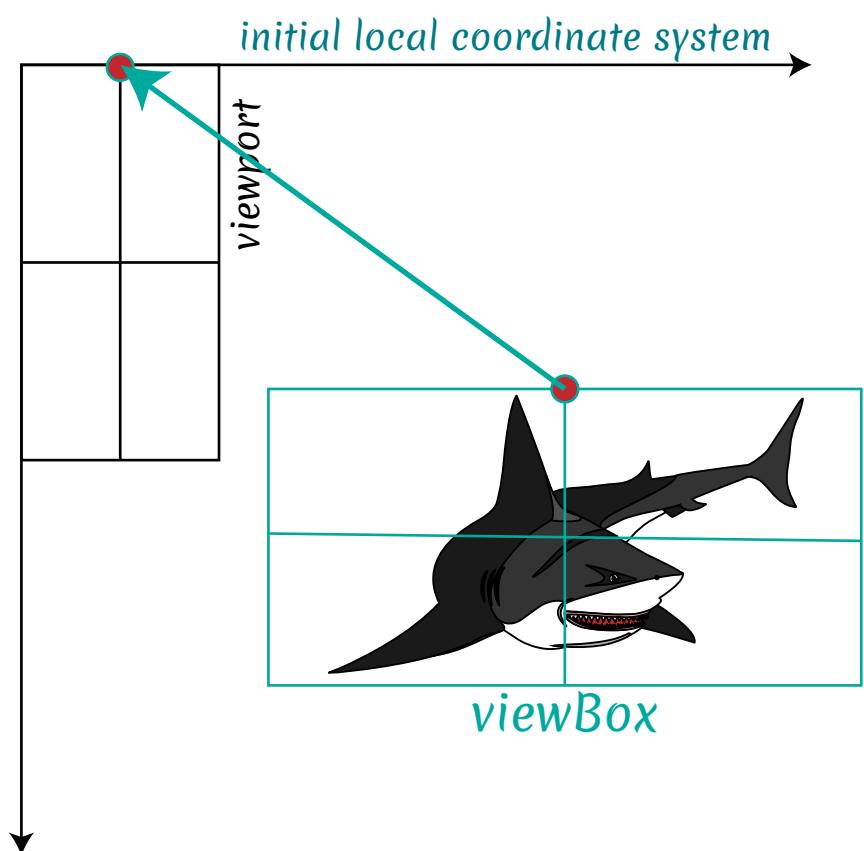


The other 9 values define the origin points of the viewBox and the viewport that will coincide on the second step before the user agent will apply the scale function. In the previous lesson, we have already used the central and left top origin points. In this lesson, we will discover them all.



lesson 4

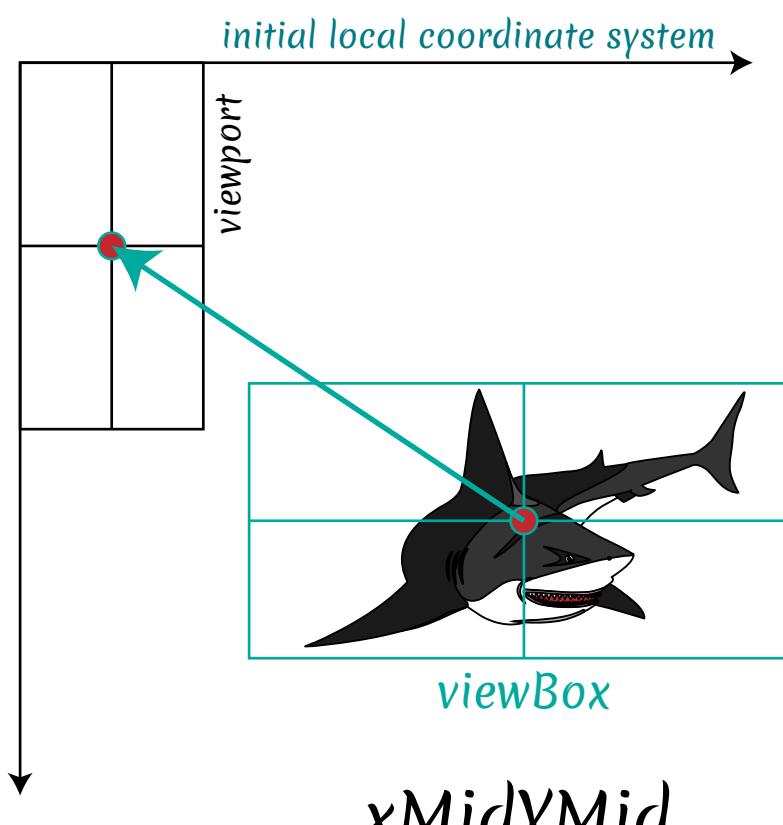
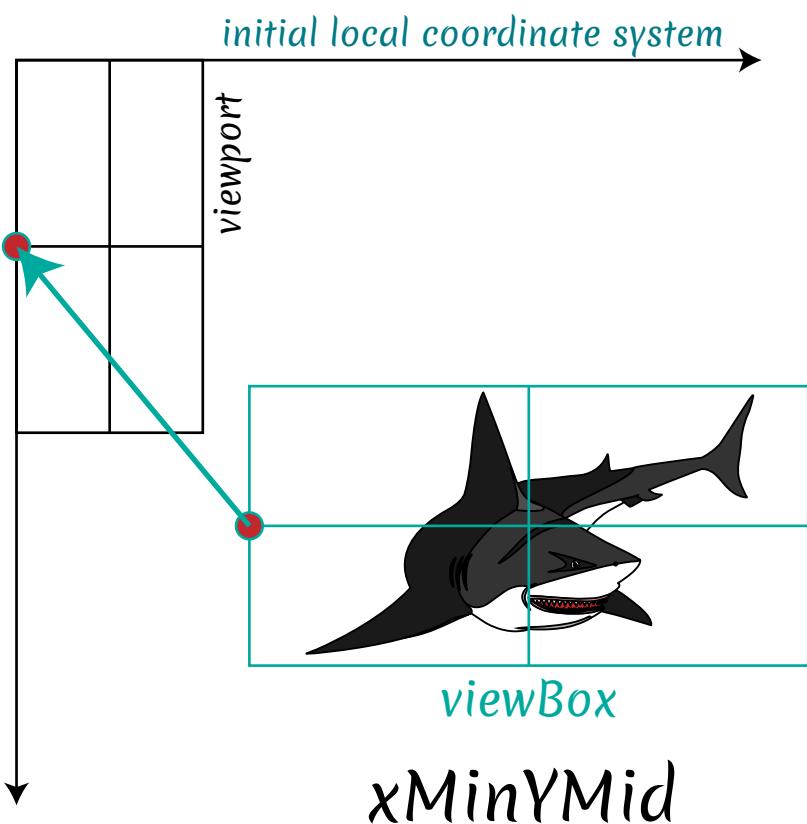
PRESERVEASPECTRATIO



xMaxYMin

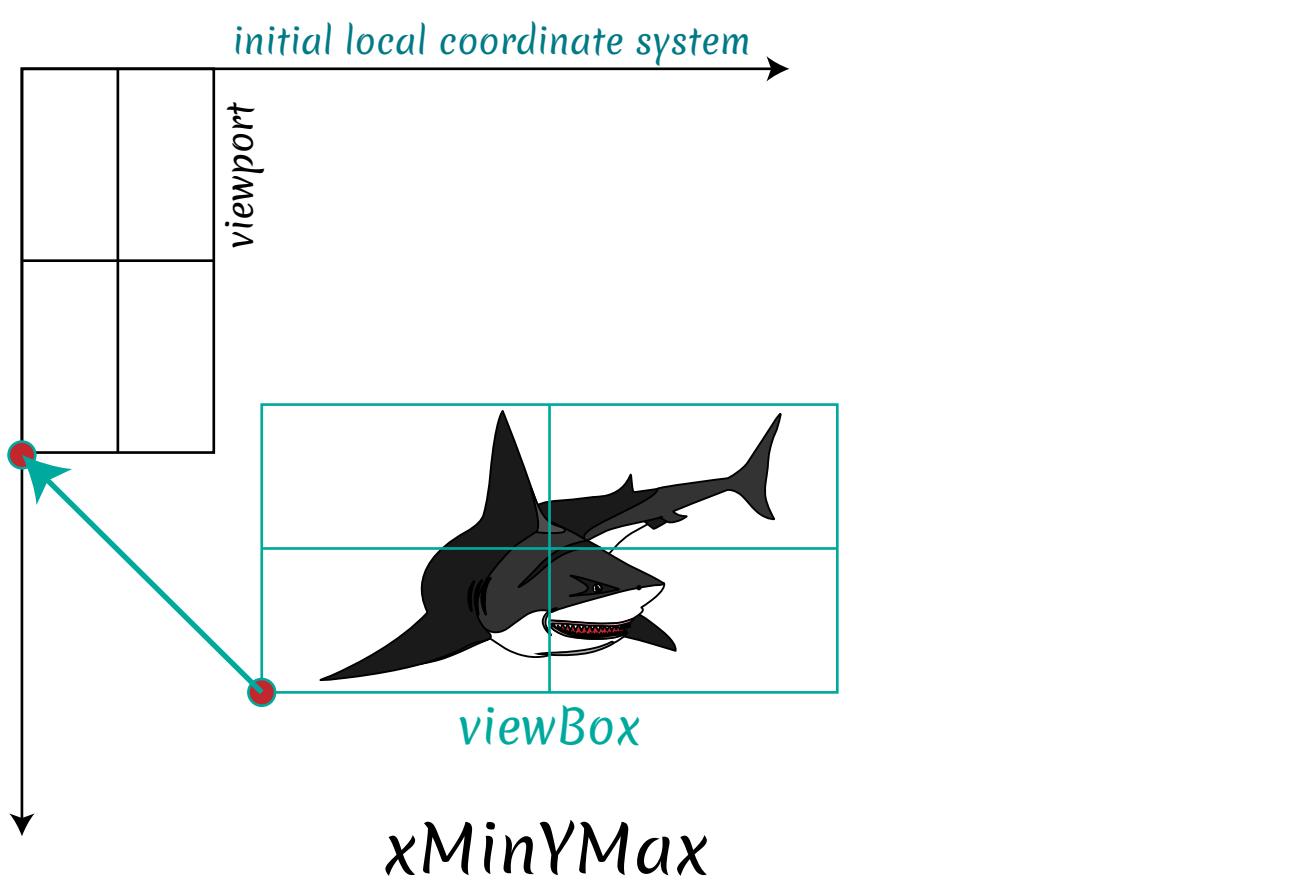
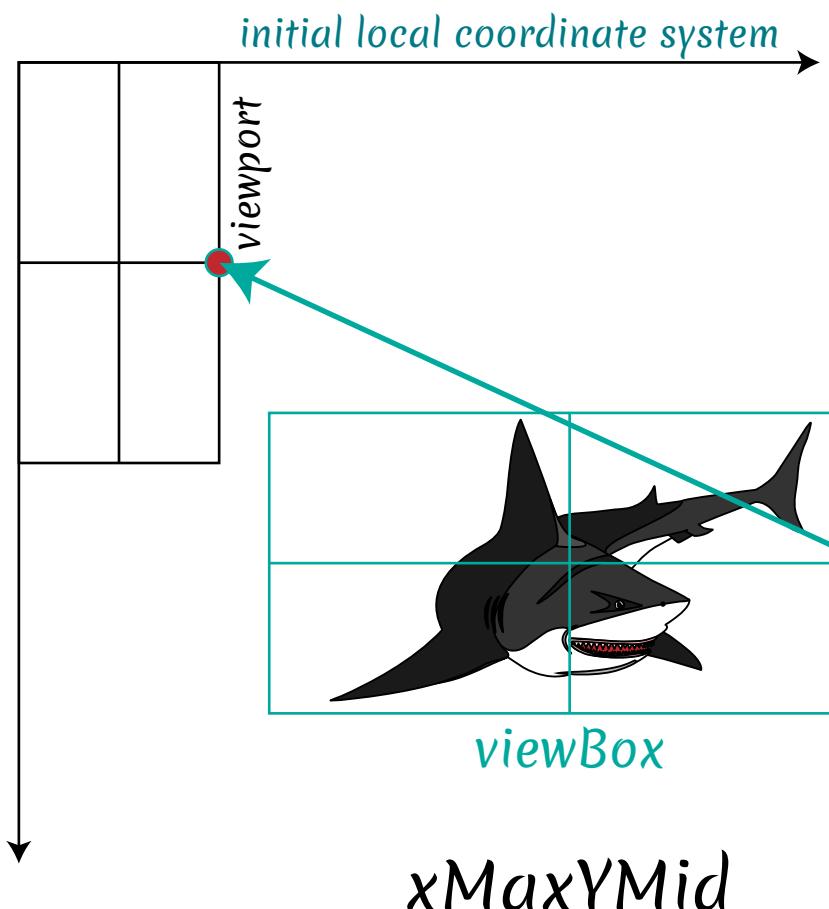
lesson 4

PRESERVEASPECTRATIO



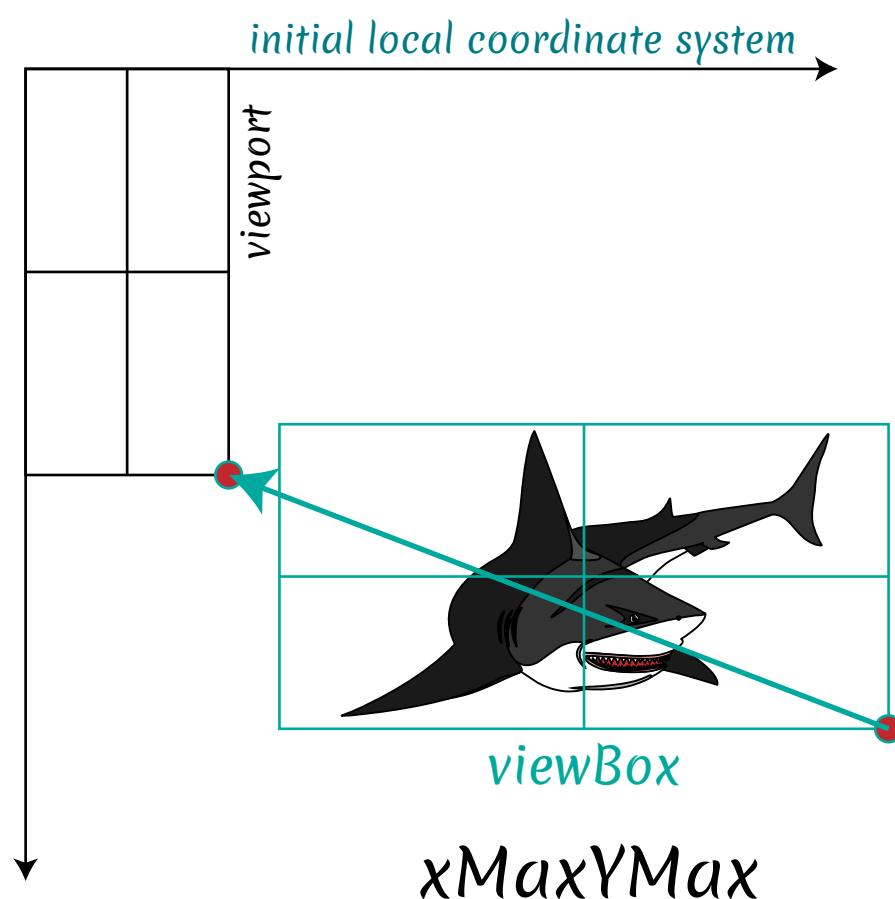
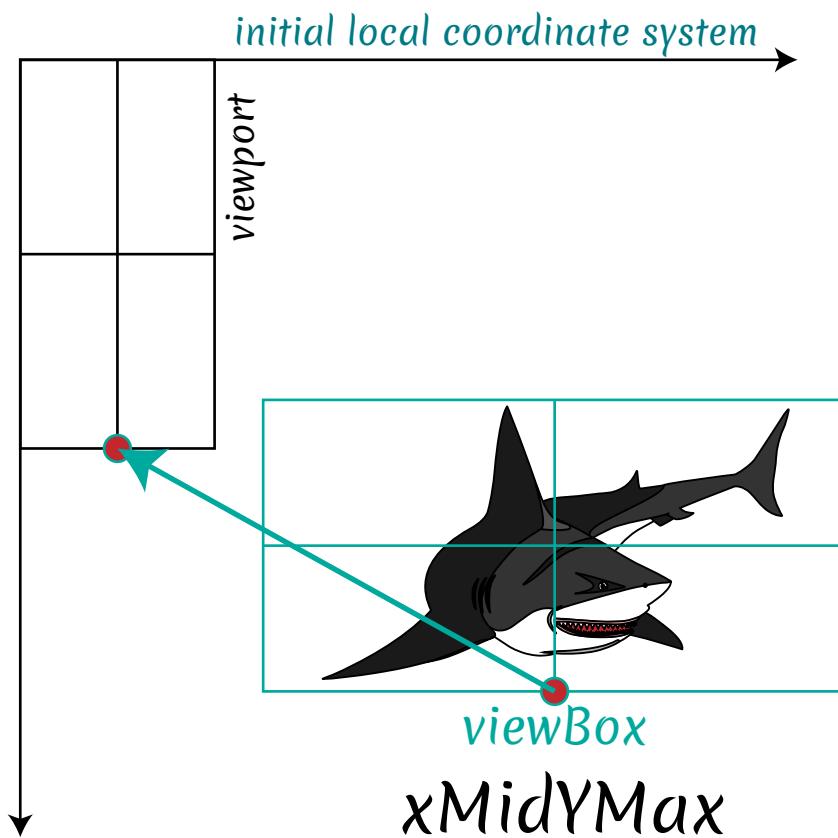
lesson 4

PRESERVEASPECTRATIO



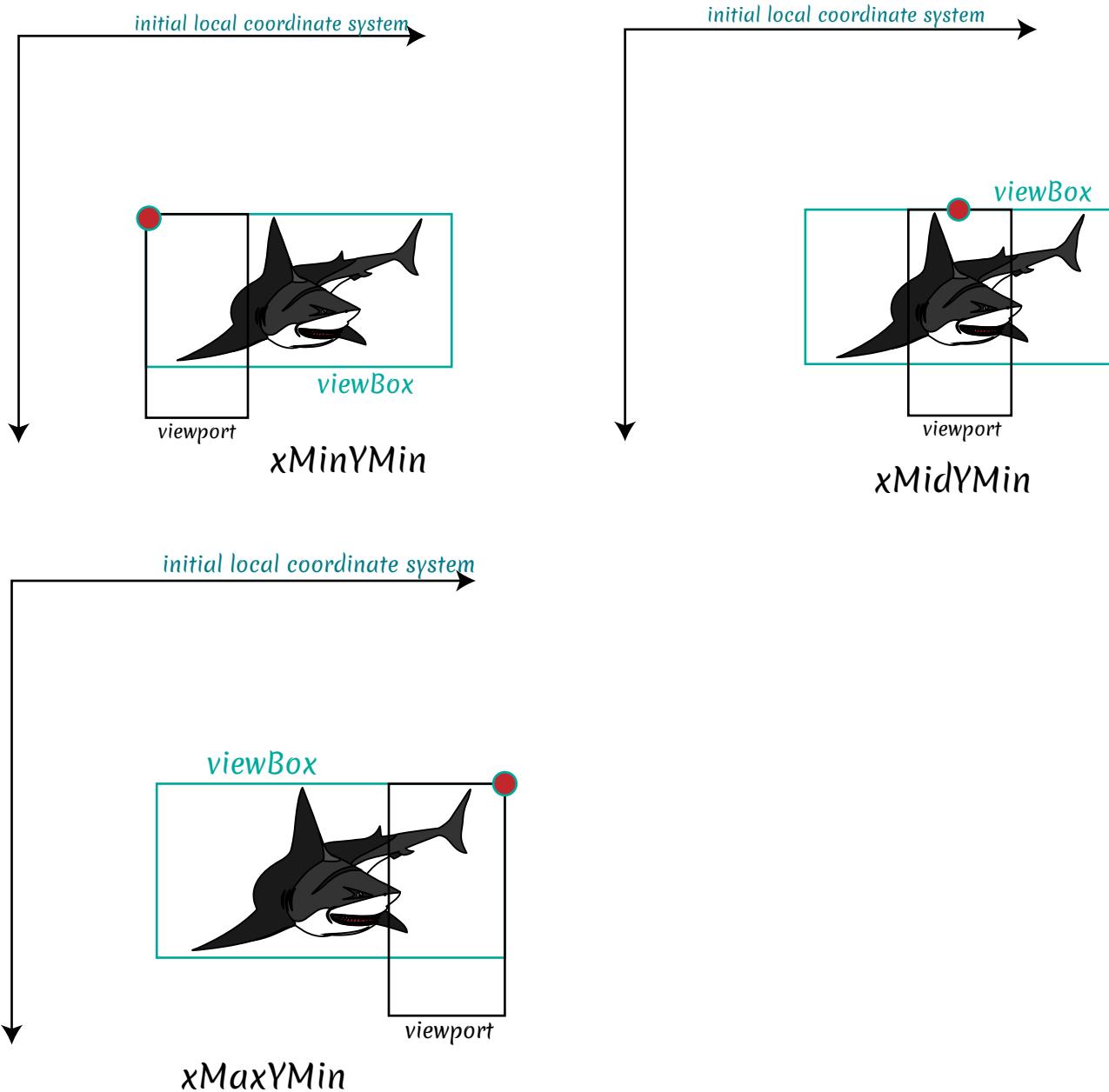
lesson 4

PRESERVEASPECTRATIO



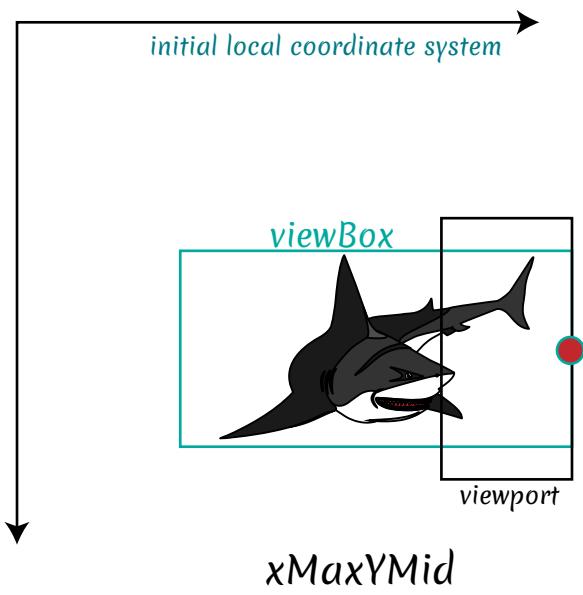
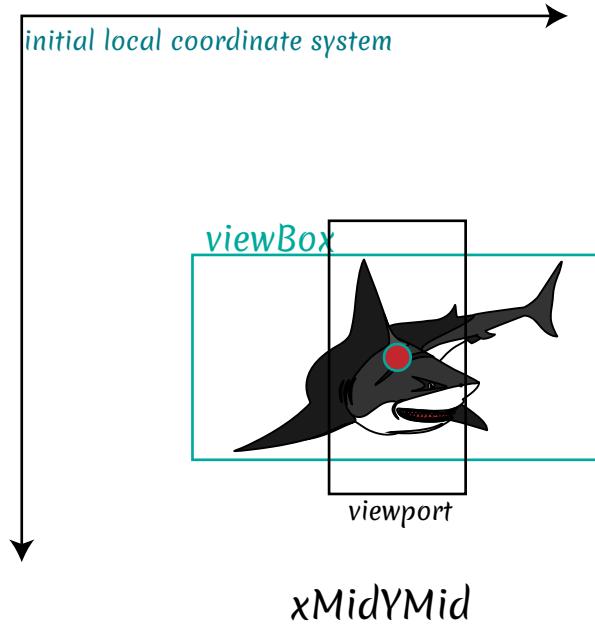
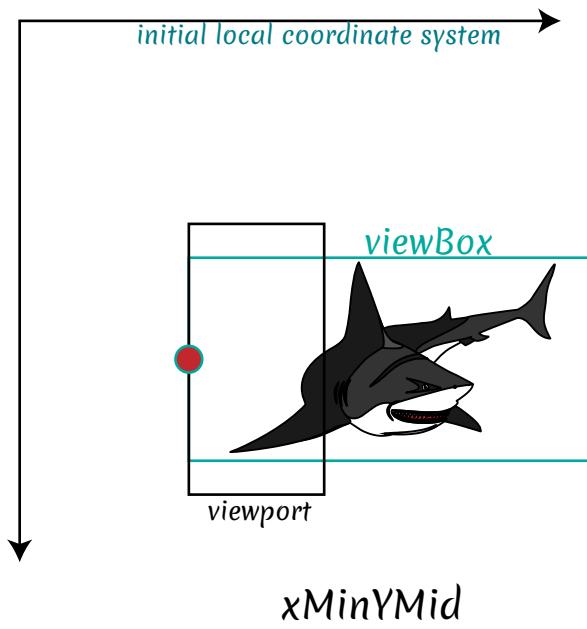
lesson 4

PRESERVEASPECTRATIO



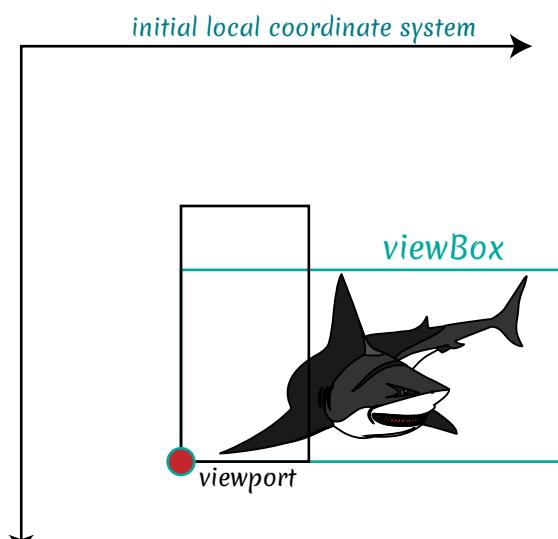
lesson 4

PRESERVEASPECTRATIO

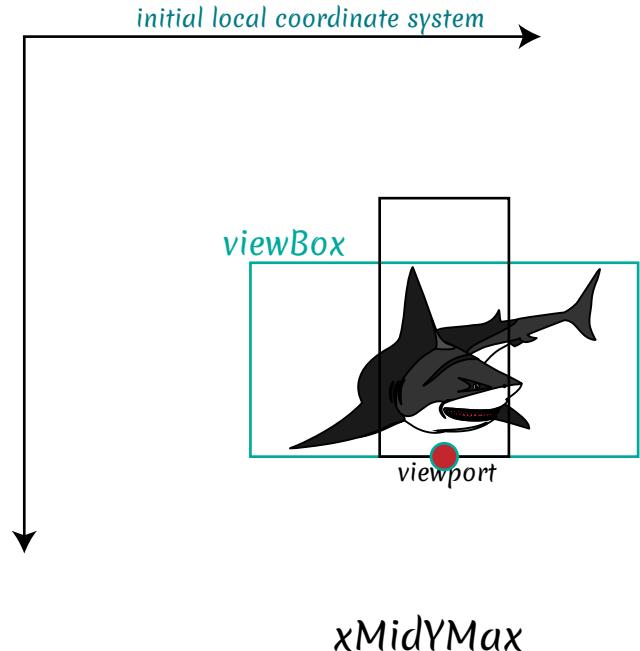


lesson 4

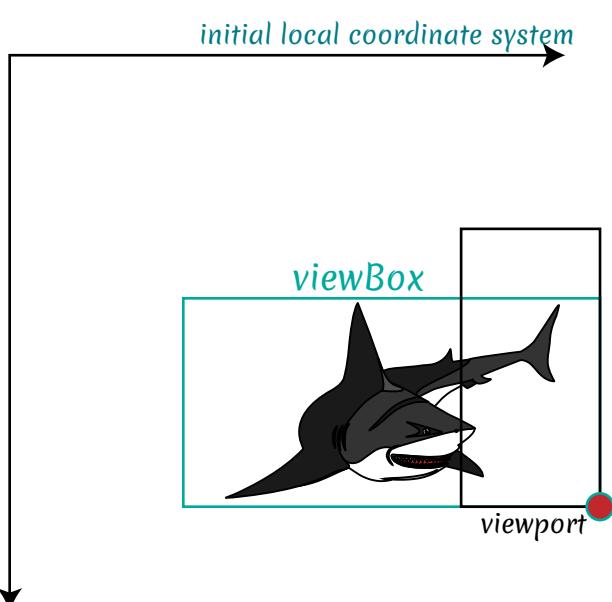
PRESERVEASPECTRATIO



xMinYMax



xMidYMax



xMaxYMax

lesson 4

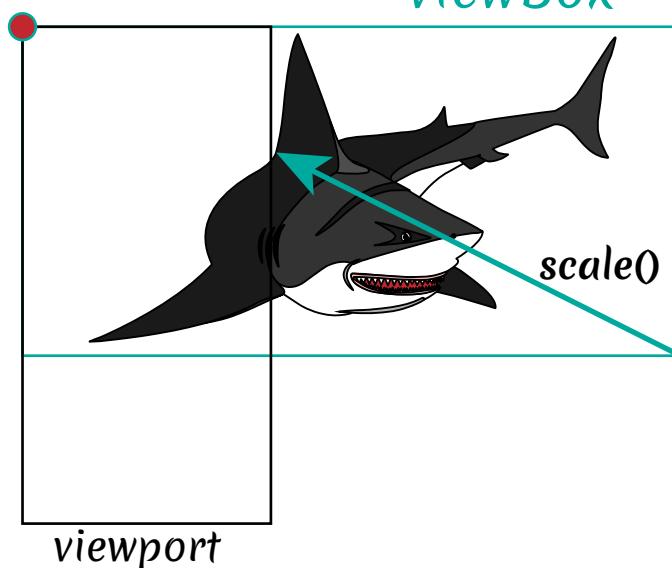
PRESERVEASPECTRATIO

meetOrSlice parameter

The meetOrSlice parameter determines what will happen in the third stage. We know what viewBox will decrease in size if the size of the viewBox is larger than the size of the viewport, and viewBox will increase if the size of the viewBox is smaller than the viewport. We know that the user agent will apply the transform=" scale()" function to do so. We know that the user agent applies scale function not only to a mask layer(viewBox layer) but also to the local coordinate system layer.

We know that viewBox should fit into the viewport. The situation was simple while the viewBox and viewport aspect ratios were the same. It didn't matter what of the <align> parameter we chose. The result was the same. The same was true for the meetOrSlice parameter. But now, the situation is very different. From now on, every value of both attributes will affect the result. We will cover every case for a meet and for slice parameters independently.

xMinYMin



3.1

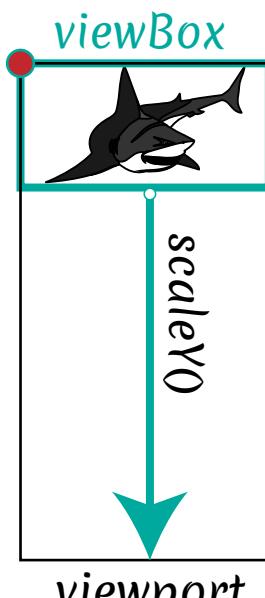
meet



3.2

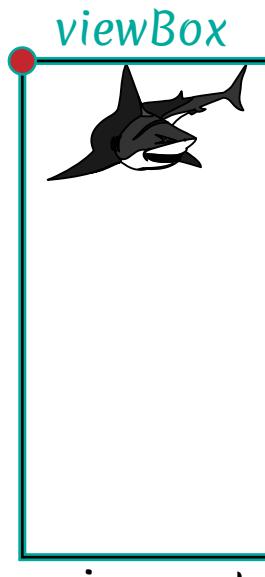
lesson 4

PRESERVEASPECTRATIO



3.3

I divided the third step into substeps you can see above. We will cover the case if the viewBox width is larger than the viewport width. If the `meetOrSlice` parameter is set inside the `preserveAspectRatio` attribute., then viewBox will fit into the viewport at step 3.1. That means that `scale` function will apply to the bound layers. So we can see that viewBox is completely fit the viewport on step 3.2. User-agent release the bound layers at step 3.3, so all following transformation will occur over the viewBox without touching the local coordinate system. We can see that the bottom side of the viewBox doesn't adjacent to the bottom side of the viewport. So we could use the phrase "boundaries meet" as a metaphor. User-agent apply a `scaleY` function to stretch the viewBox to let its bottom side meet with the bottom side of the viewport. So the size and aspect ratio of the viewBox is changing, and that means that if there were graphics below the shark on the local coordinate system it become partially visible.

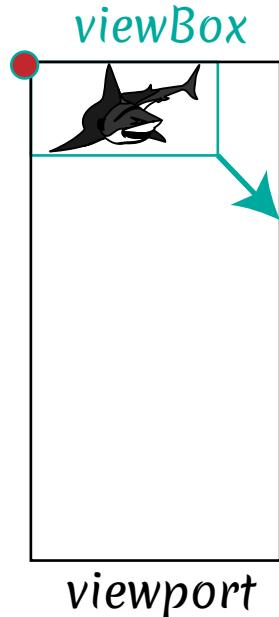


3.4

lesson 4

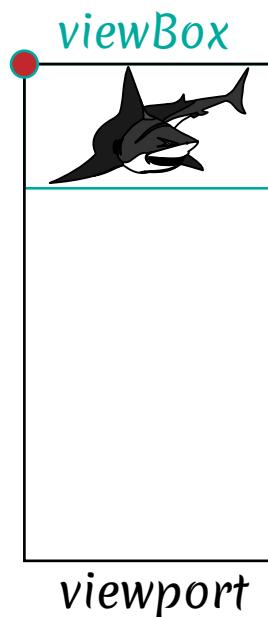
PRESERVEASPECTRATIO

xMinYMin



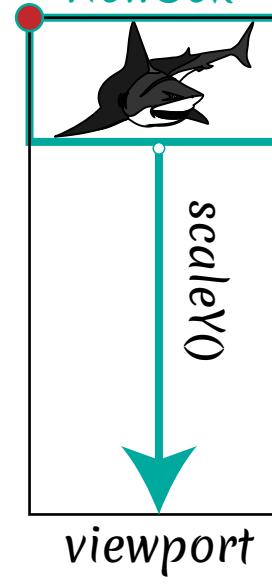
3.1

meet



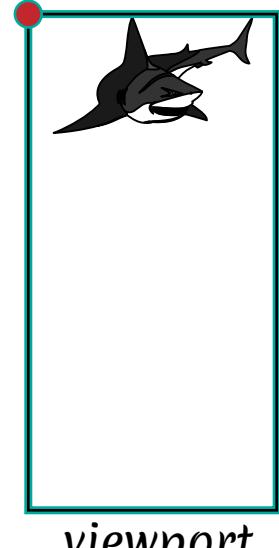
3.2

viewBox



3.3

viewBox



3.4

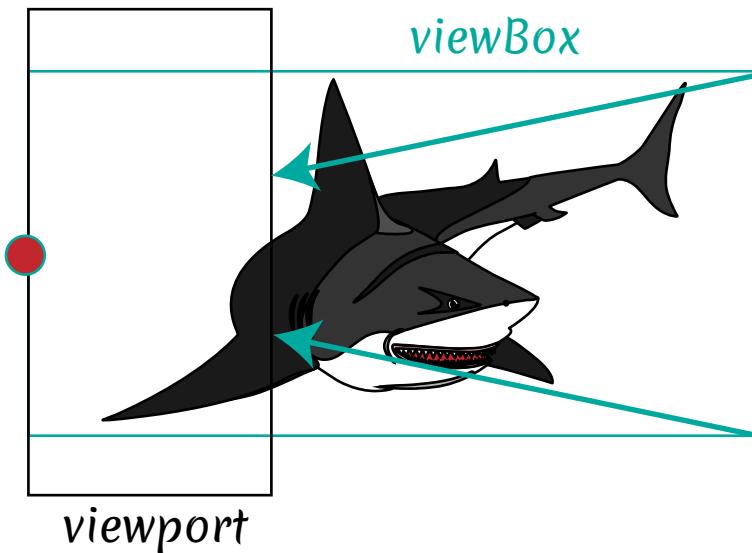
In case, if the viewBox width is smaller than the viewport width and the viewport positioned vertically, at step 3.1 user-agent increase the size of the bound layers by applying the scale function with the value greater than 1. The other substeps are identical.

lesson 4

PRESERVEASPECTRATIO

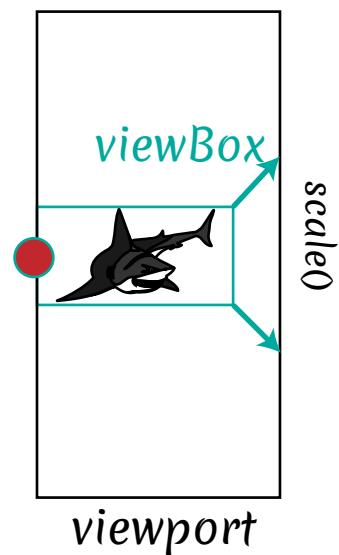
xMinYMid

meet

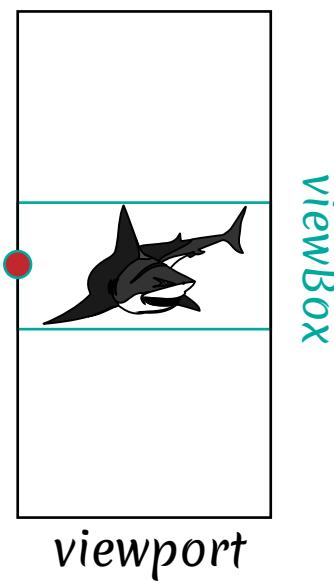


3.1

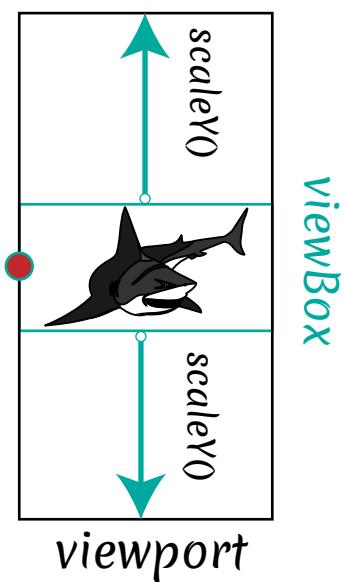
OR



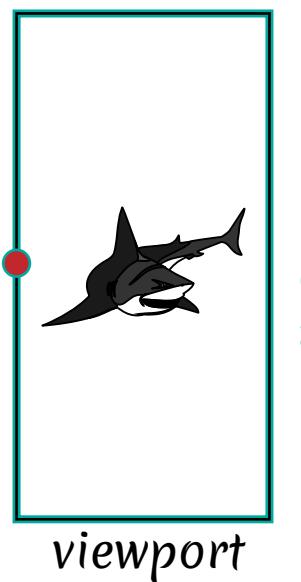
3.1



3.2



3.3



3.4

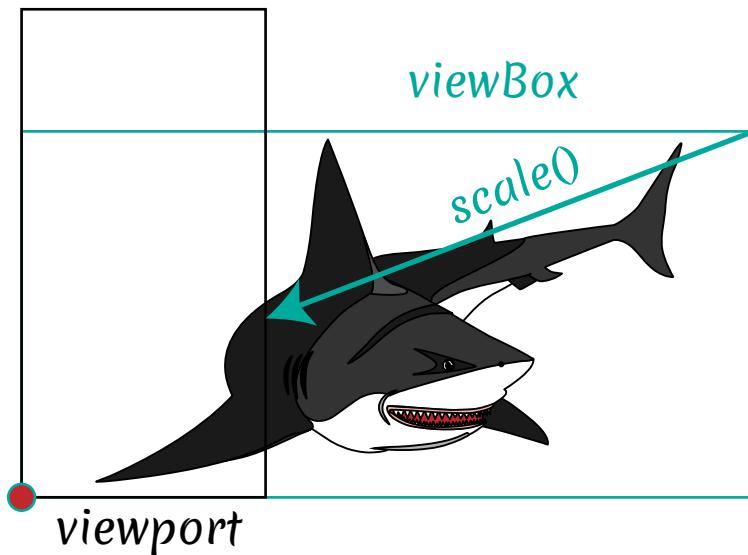
viewport

lesson 4

PRESERVEASPECTRATIO

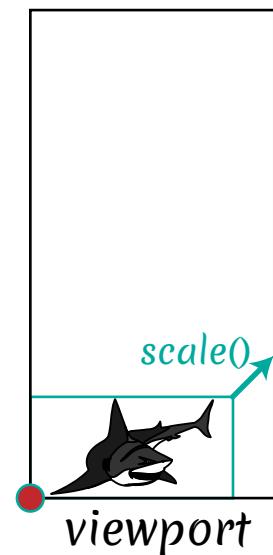
xMinYMax

meet

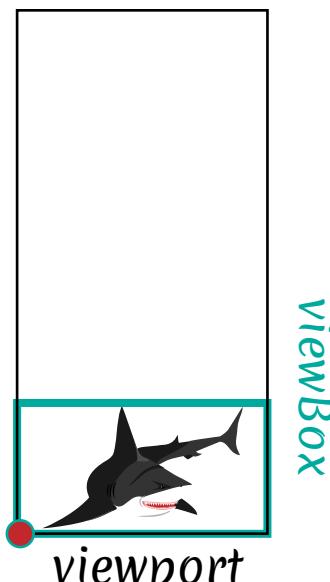


3.1

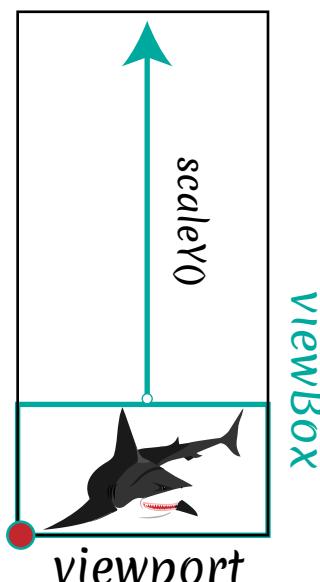
OR



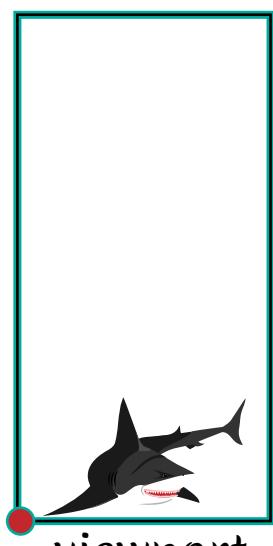
3.1



3.2



3.3



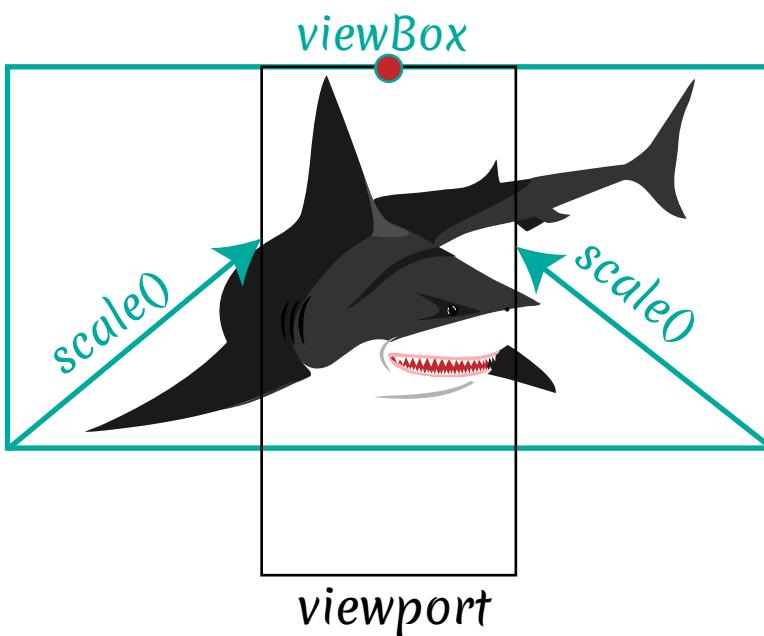
3.4

viewport

lesson 4

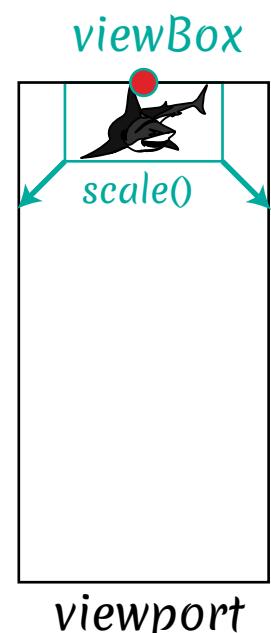
PRESERVEASPECTRATIO

xMidYMin



meet

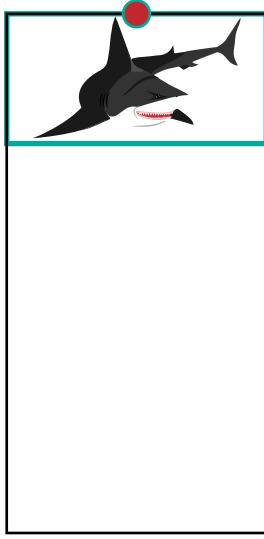
OR



3.1

3.1

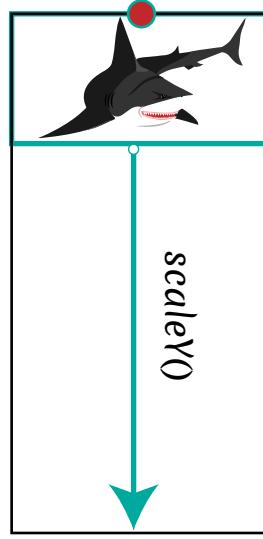
viewBox



viewport

3.2

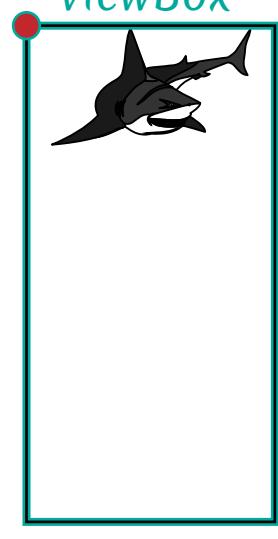
viewBox



viewport

3.3

viewBox



viewport

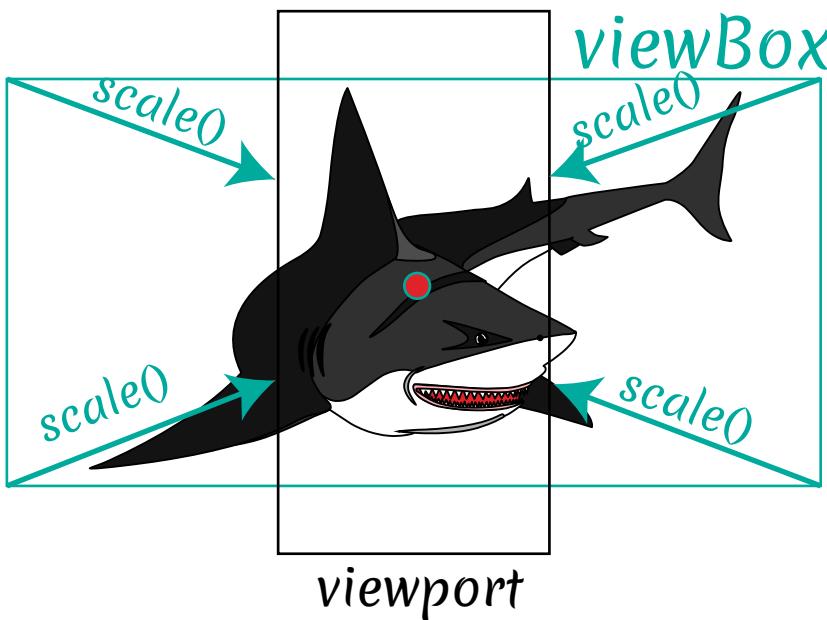
3.4

lesson 4

PRESERVEASPECTRATIO

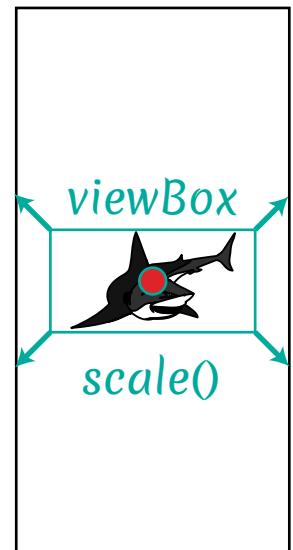
xMidYMid

meet



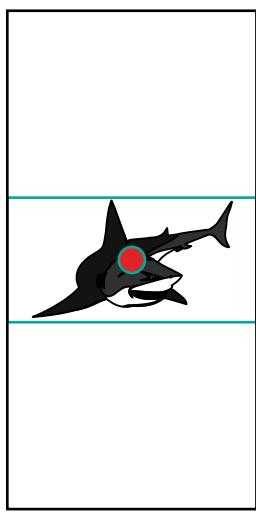
3.1

OR



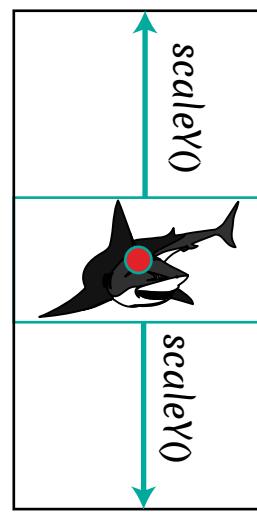
viewport

3.1



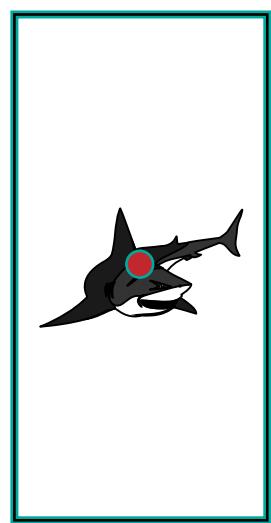
viewport

3.2



viewport

3.3



viewport

3.4

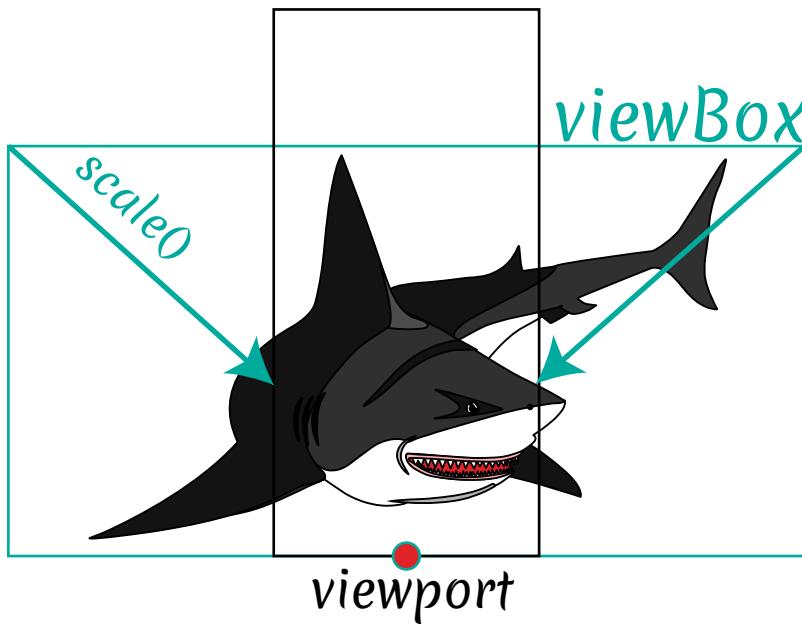
viewBox

lesson 4

PRESERVEASPECTRATIO

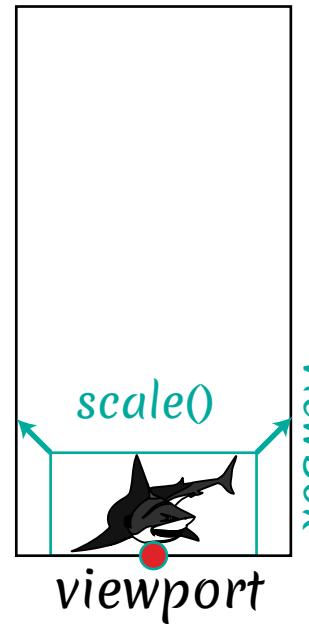
xMidYMax

meet

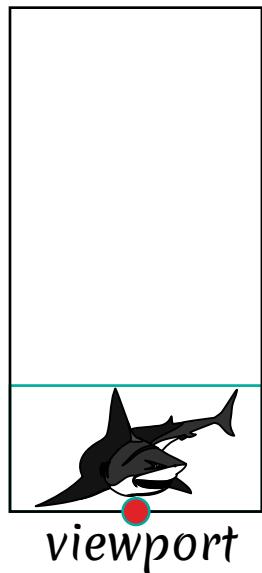


3.1

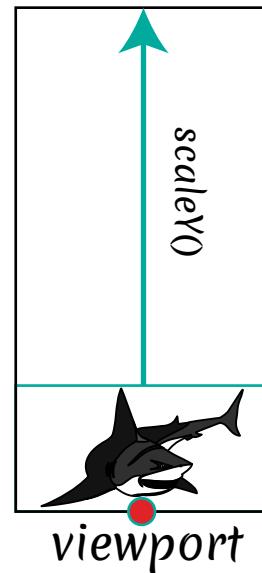
OR



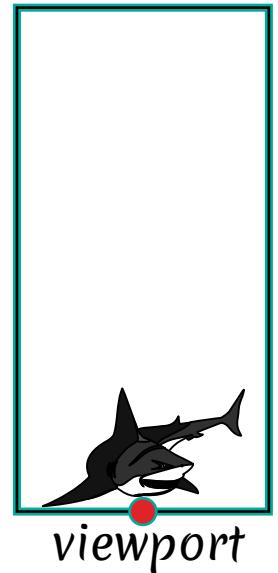
3.1



3.2



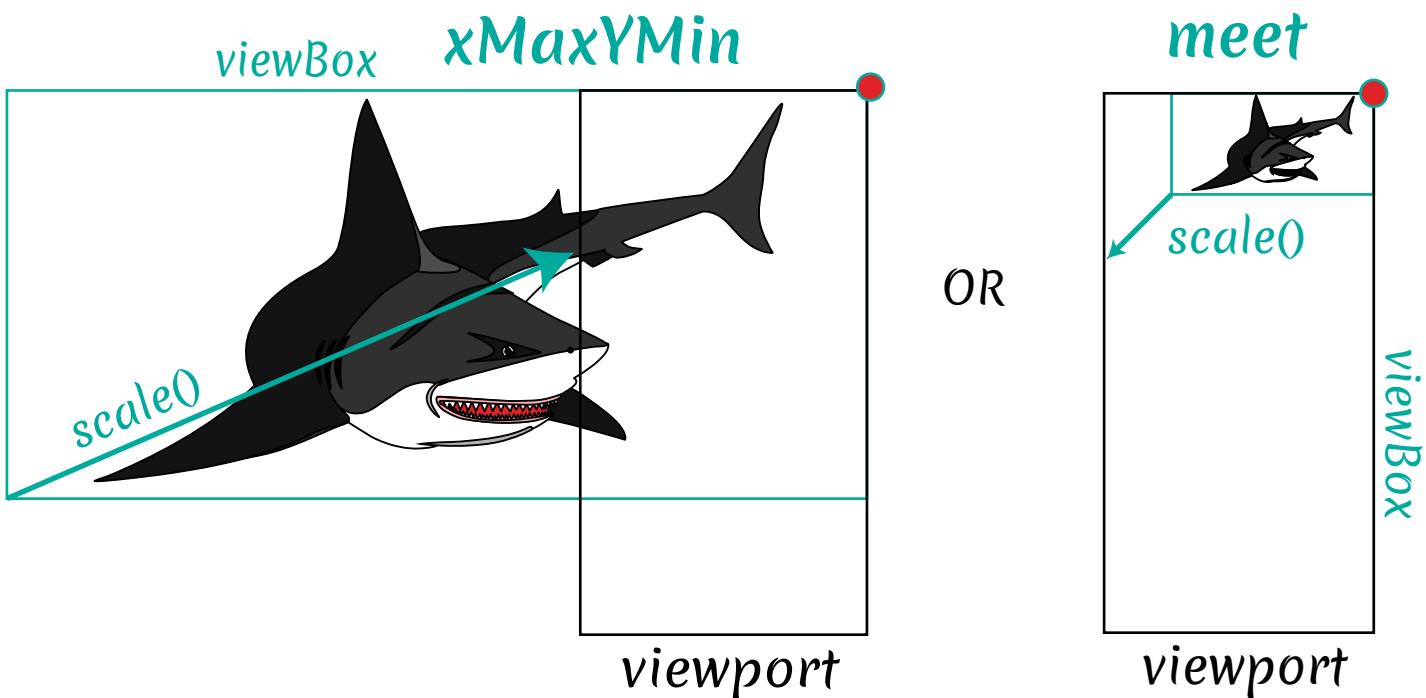
3.3



3.4

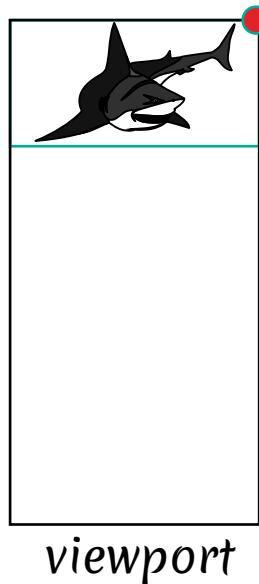
lesson 4

PRESERVEASPECTRATIO

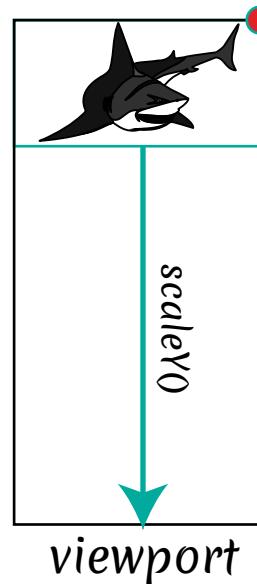


3.1

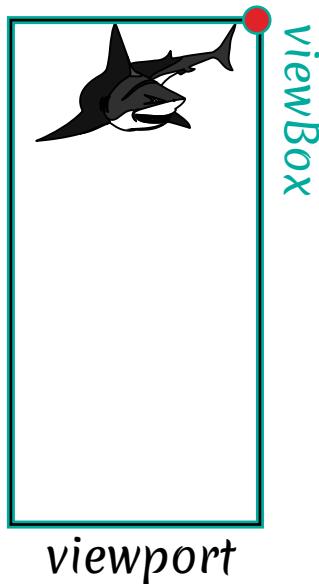
3.1



3.2



3.3



3.4

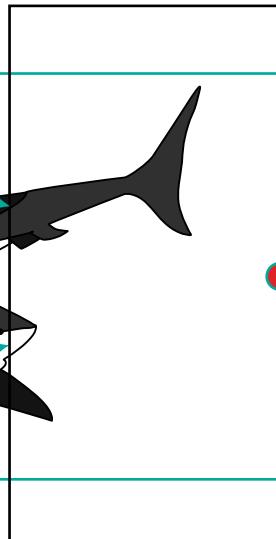
lesson 4

PRESERVEASPECTRATIO

xMaxYMid

viewBox

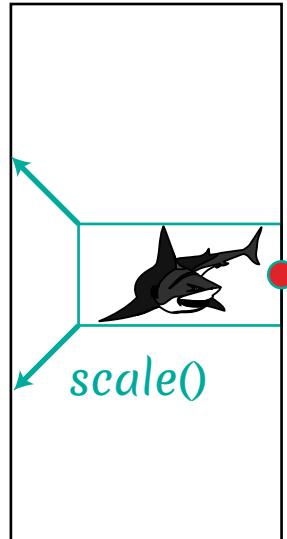
scaled0
scaled0



viewport

meet

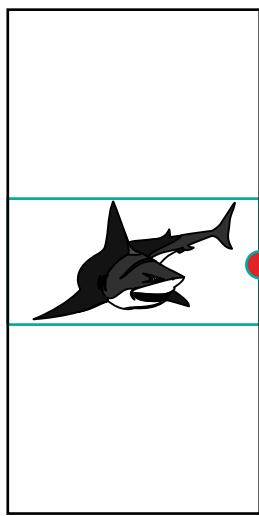
viewBox



viewport

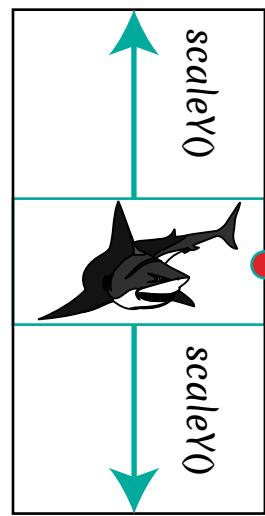
3.1

3.1



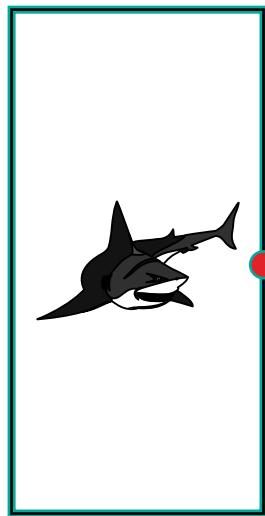
viewport

viewBox



viewport

viewBox



viewport

3.2

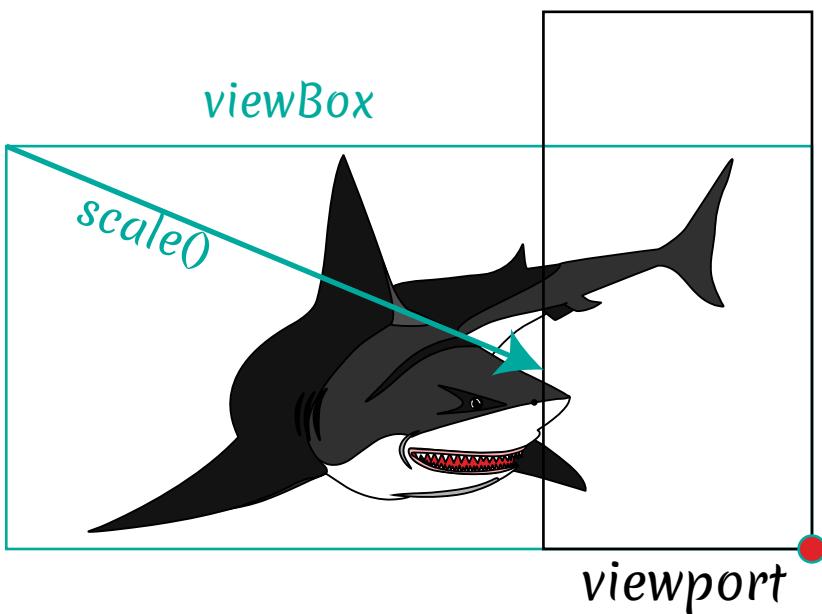
3.3

3.4

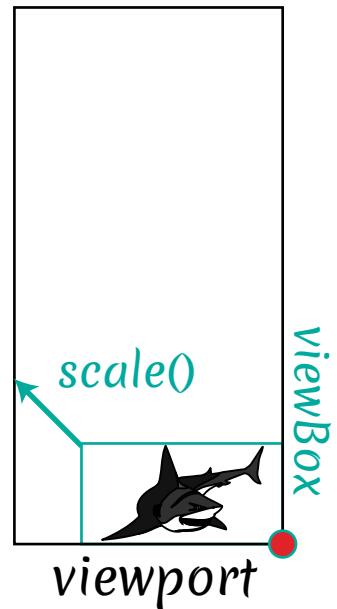
lesson 4

PRESERVEASPECTRATIO

xMaxYMax

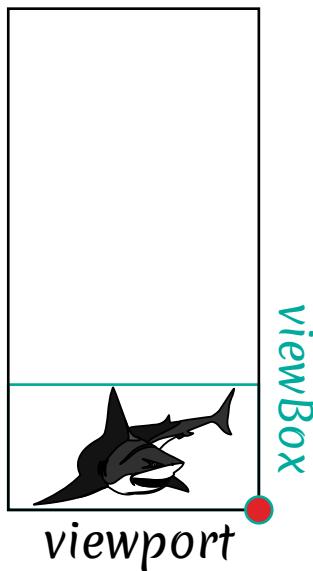


meet

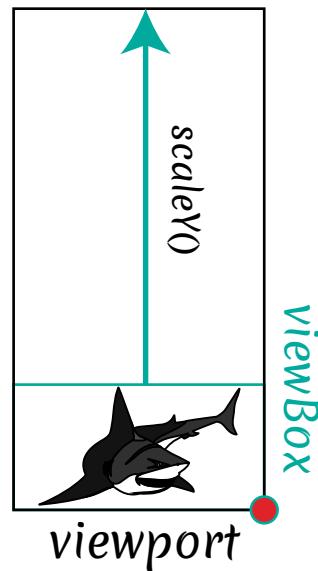


3.1

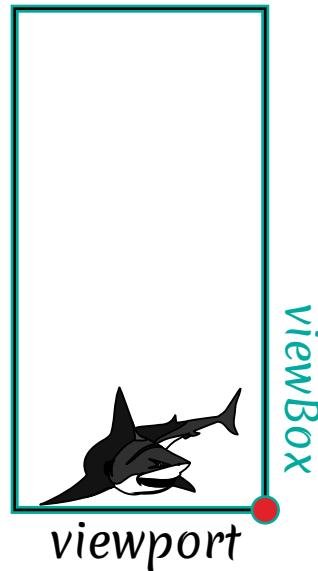
3.1



3.2



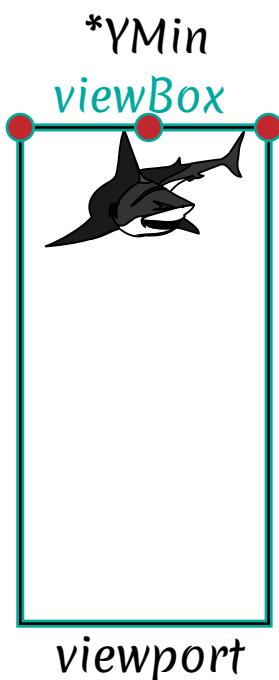
3.3



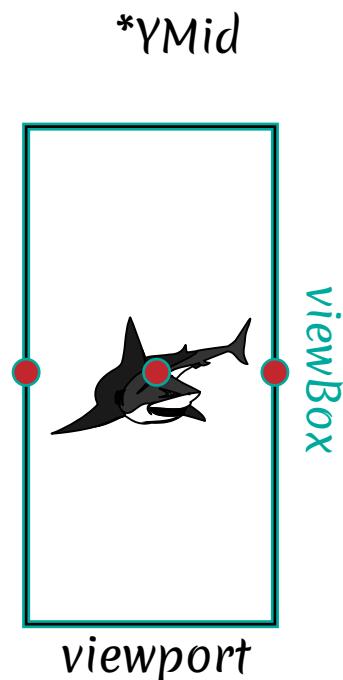
3.4

lesson 4

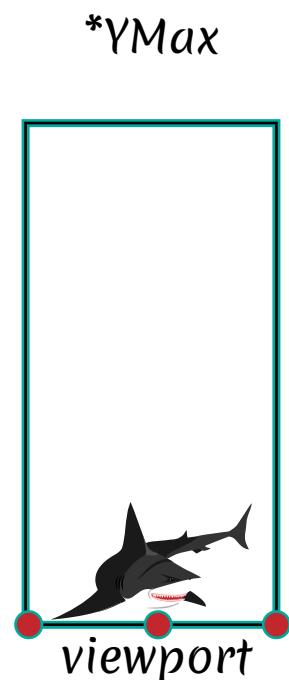
PRESERVEASPECTRATIO



$x\text{Min}y\text{Min}$
 $x\text{Mid}y\text{Min}$
 $x\text{Max}y\text{Min}$



$x\text{Min}y\text{Mid}$
 $x\text{Mid}y\text{Mid}$
 $x\text{Max}y\text{Mid}$



$x\text{Min}y\text{Max}$
 $x\text{Mid}y\text{Max}$
 $x\text{Max}y\text{Max}$

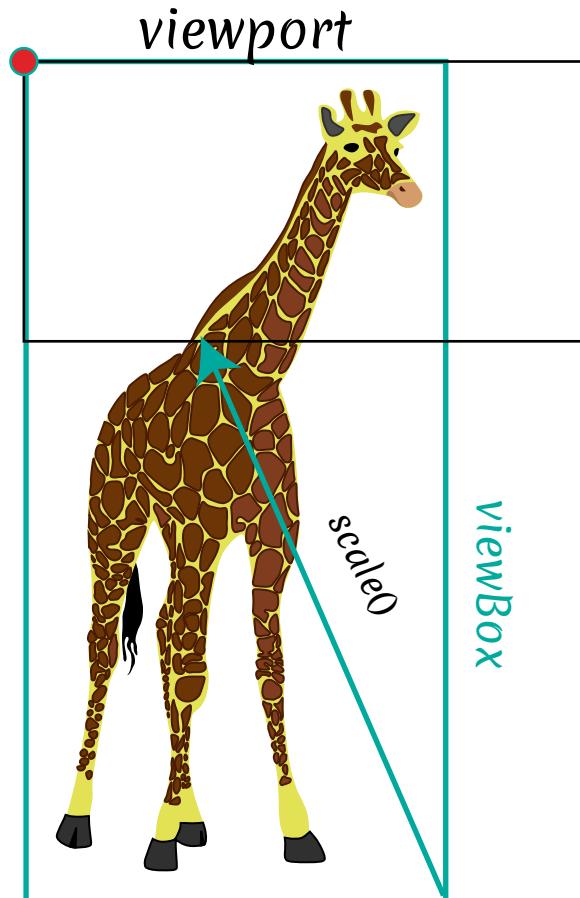
As you may have noticed, If the viewport is vertical and the viewBox is horizontal, the alignment is vertical. So it doesn't matter what xM^{**} value we use. The result will be the same.

lesson 4

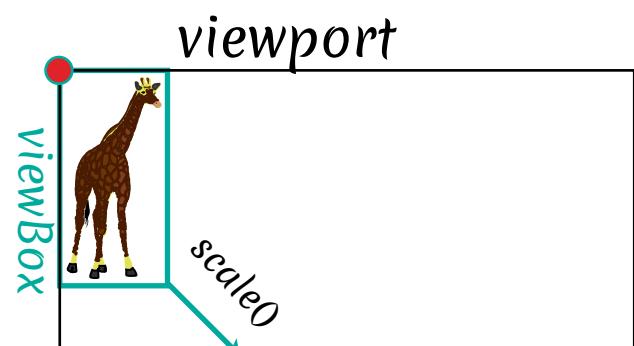
PRESERVEASPECTRATIO

xMinYMin

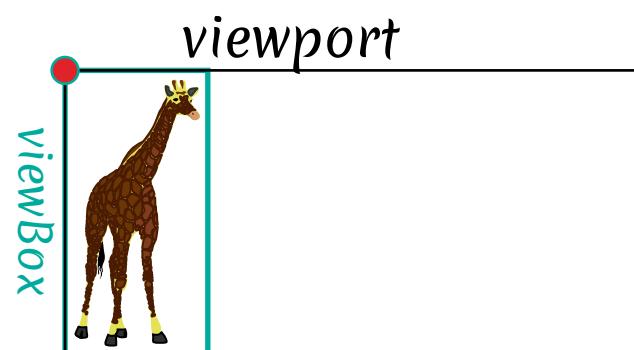
meet



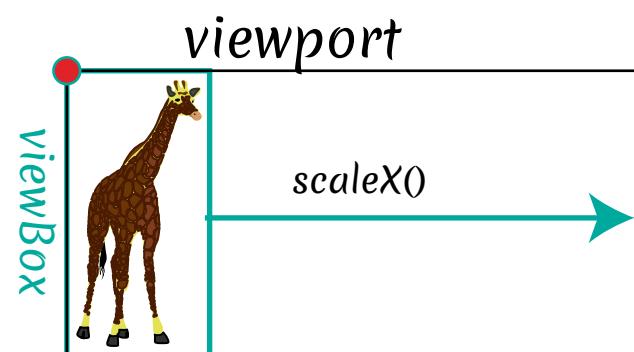
OR



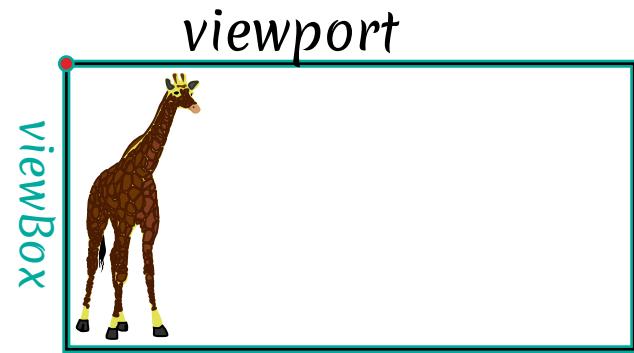
3.1



3.2



3.3



viewport

3.4

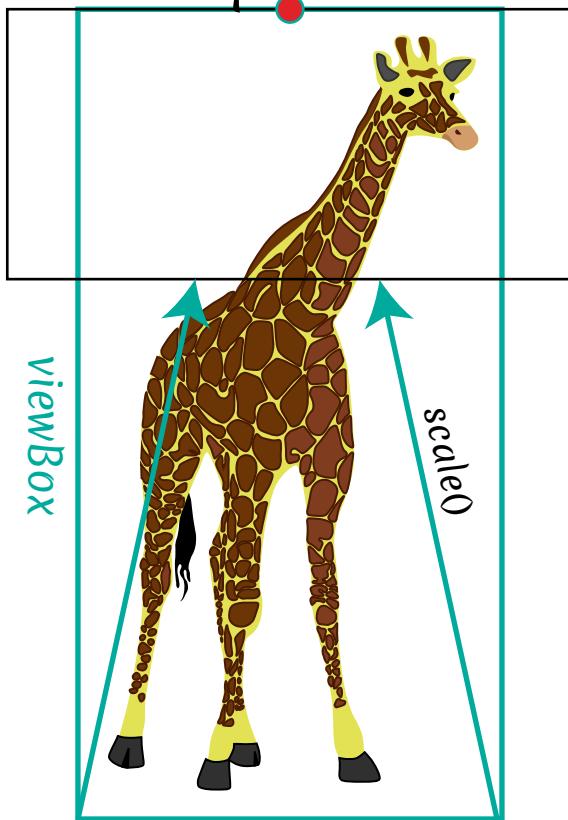
lesson 4

PRESERVEASPECTRATIO

xMidYMin

meet

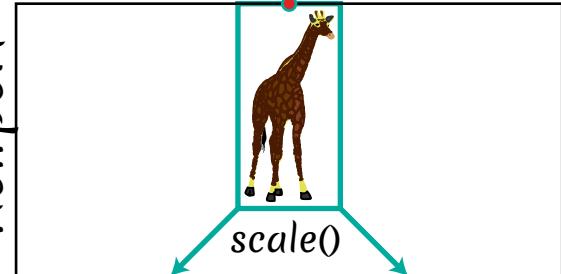
viewport



3.1

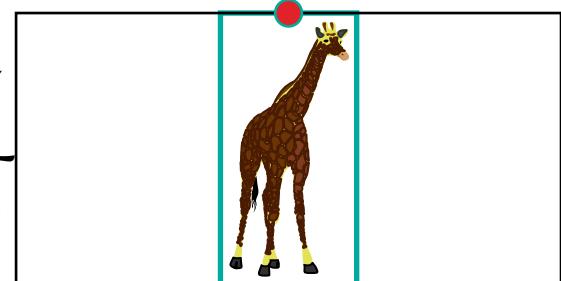
OR

viewport



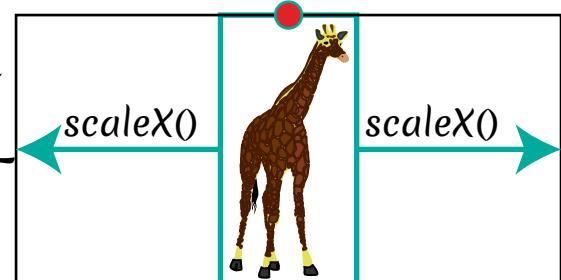
3.1

viewport



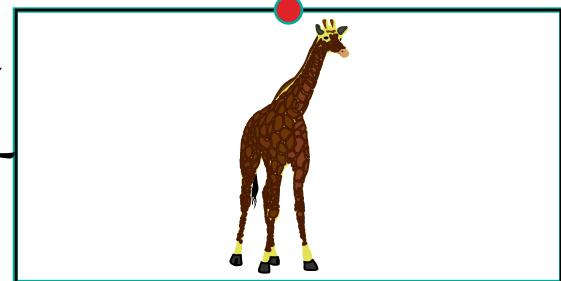
3.2

viewport



3.3
viewport

viewport



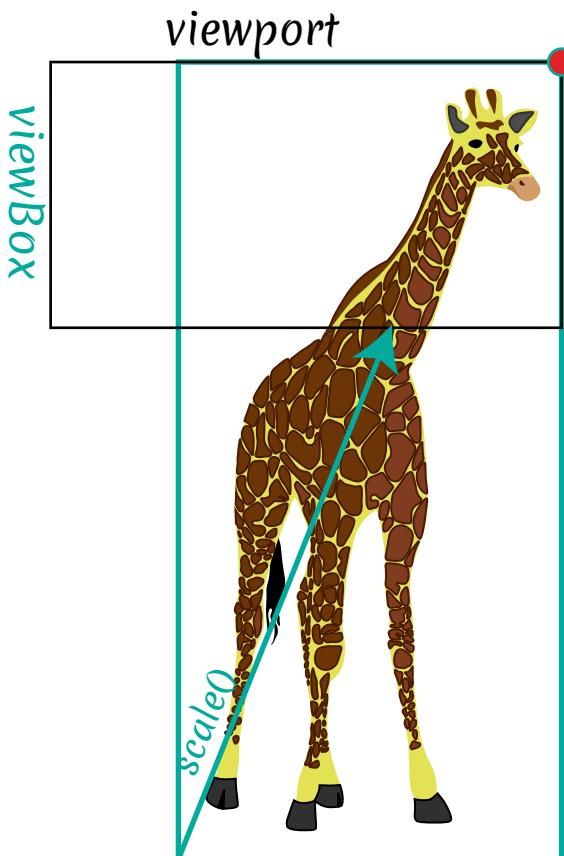
3.4

lesson 4

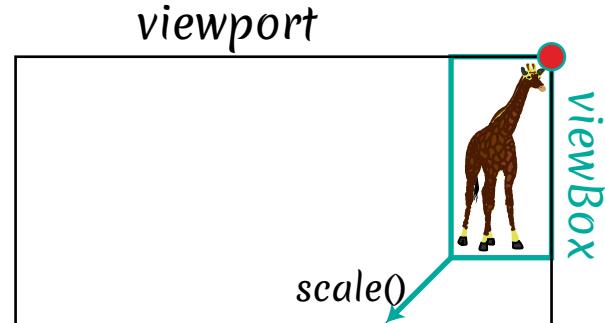
PRESERVEASPECTRATIO

xMaxYMin

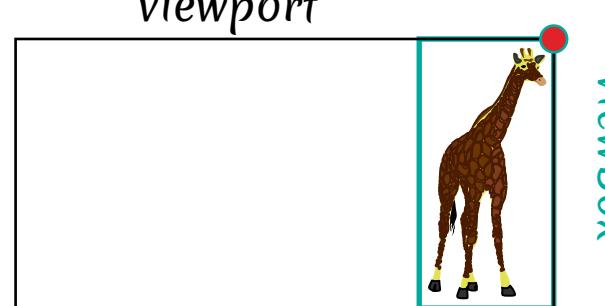
meet



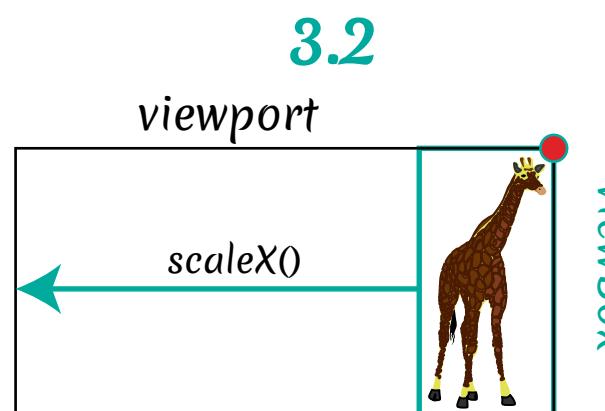
OR



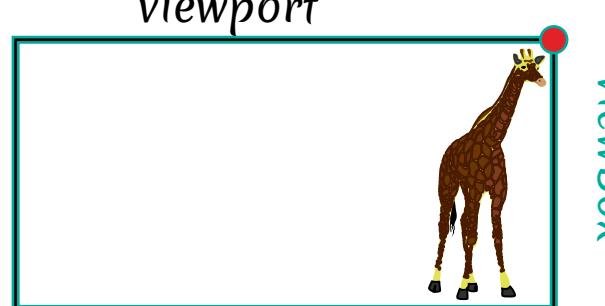
3.1



viewport



3.2



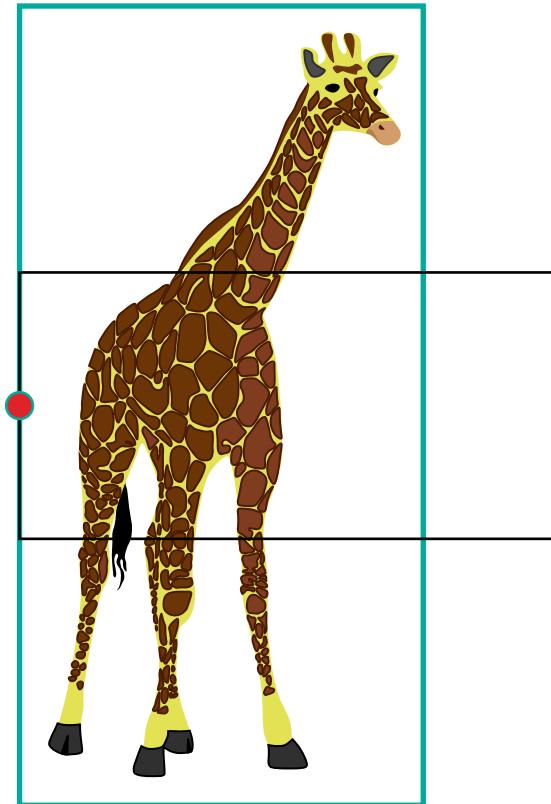
viewport

3.4

lesson 4

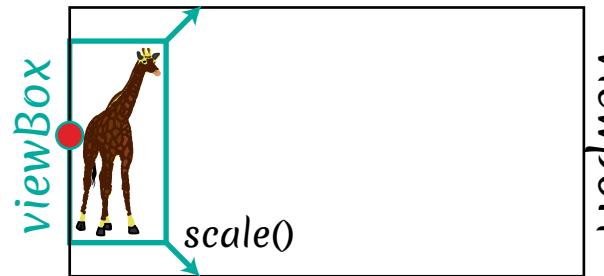
PRESERVEASPECTRATIO

viewBox

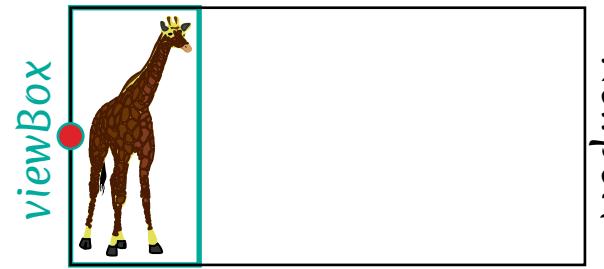


3.1

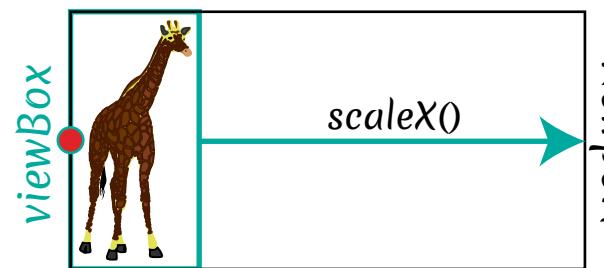
OR



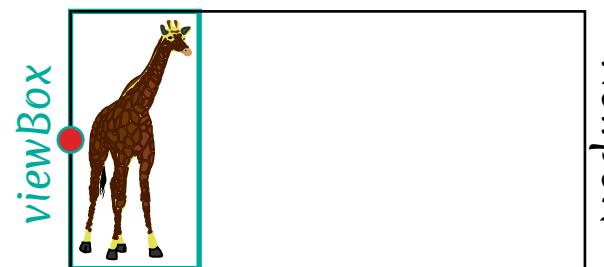
3.1



3.2



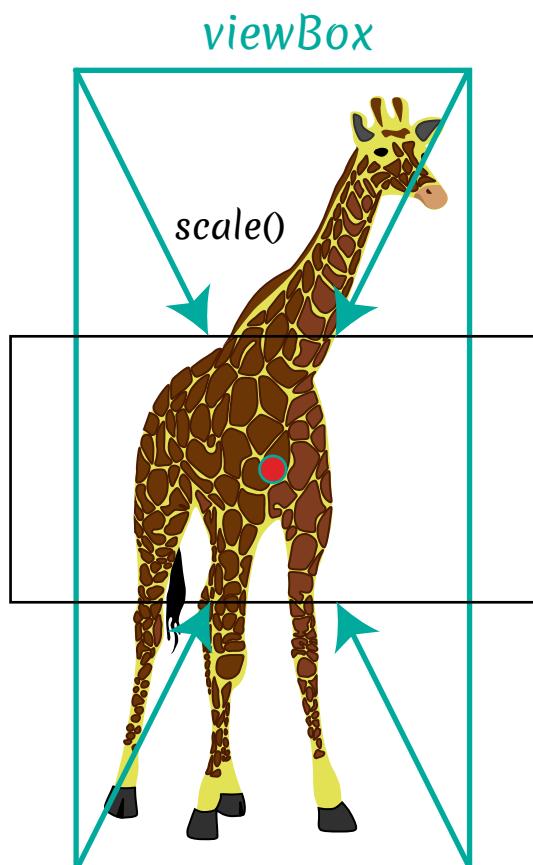
3.3



3.4

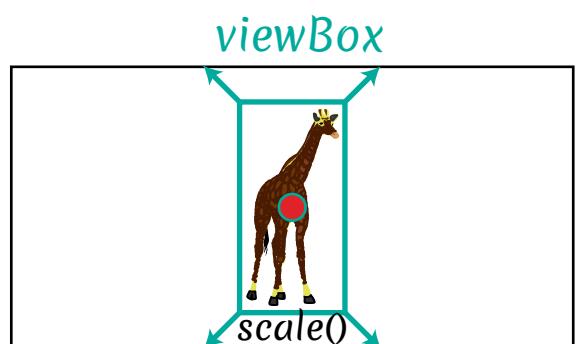
lesson 4

PRESERVEASPECTRATIO

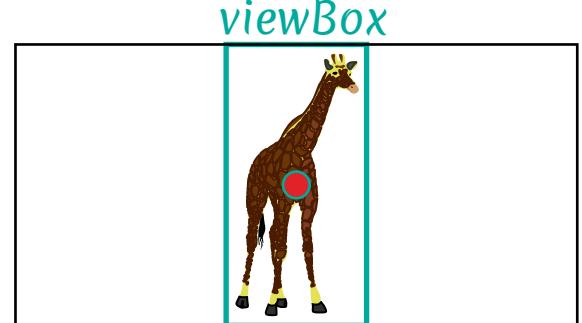


3.1

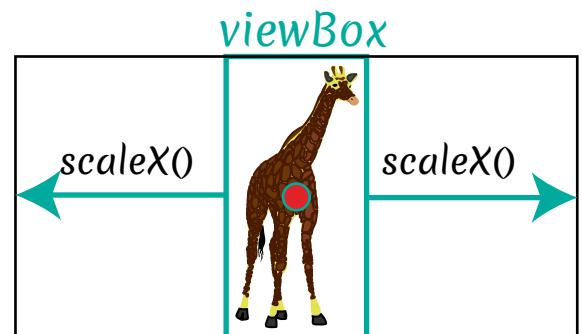
OR



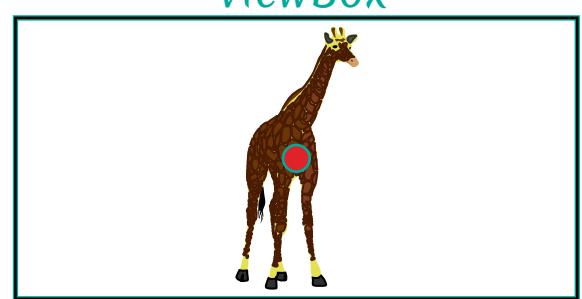
3.1



3.2



3.3

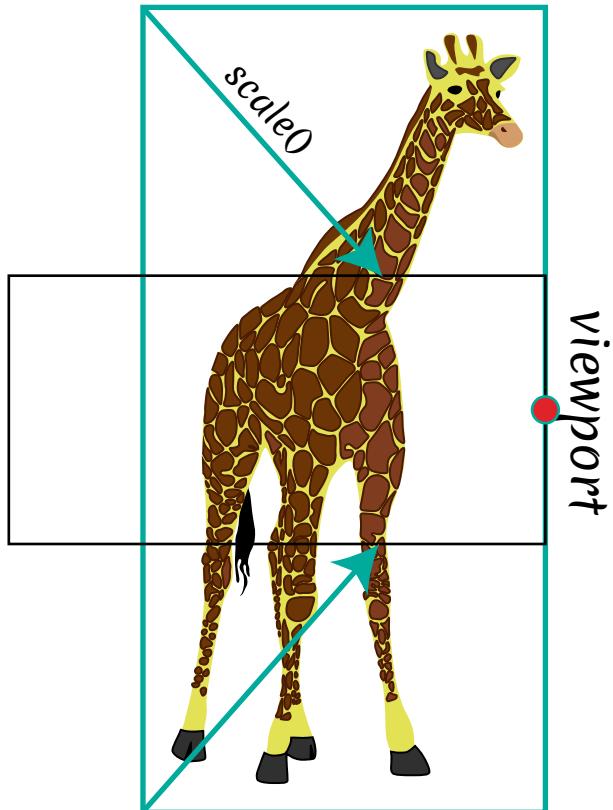


3.4

lesson 4

PRESERVEASPECTRATIO

viewBox



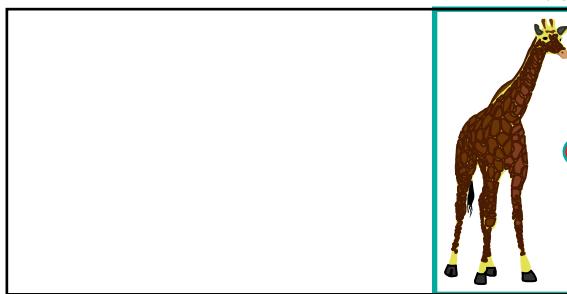
3.1

OR



3.1

viewBox



3.2

viewBox



3.3

viewBox

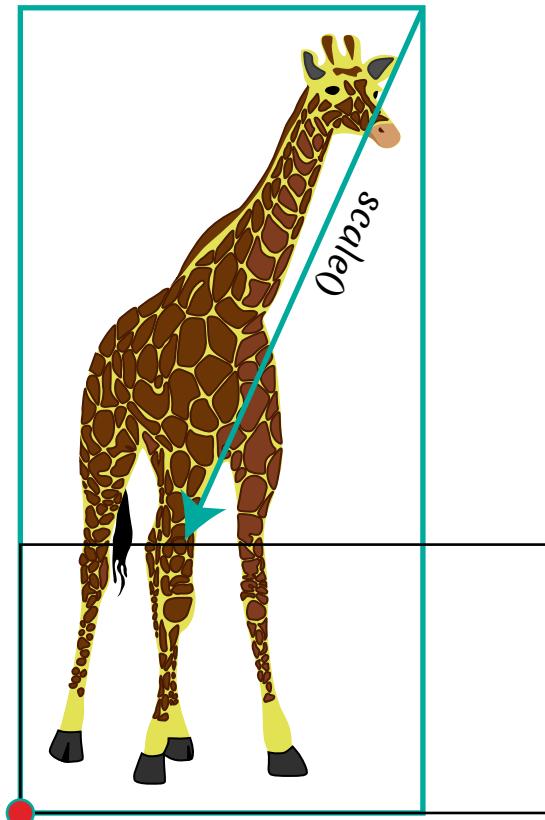


3.4

lesson 4

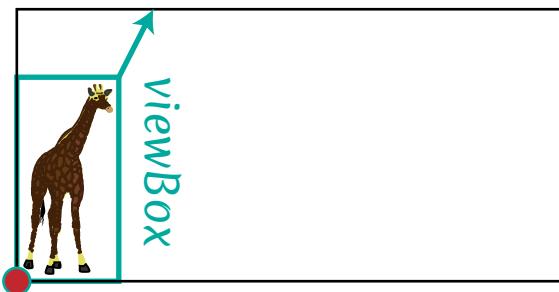
PRESERVEASPECTRATIO

viewBox



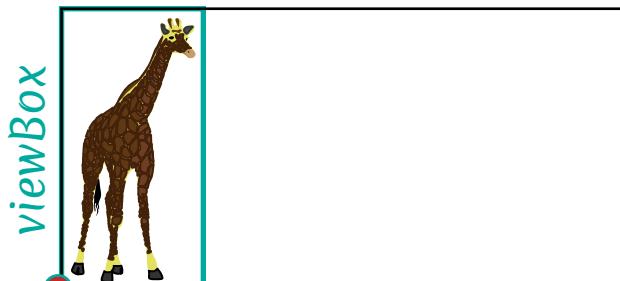
3.1

OR

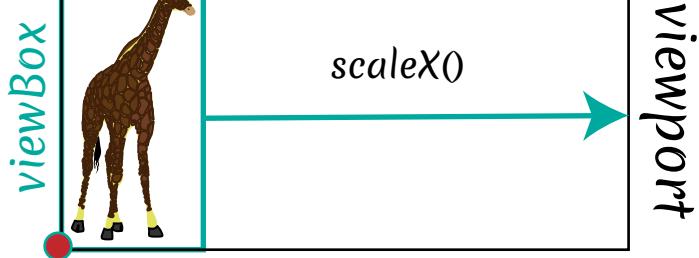


3.1

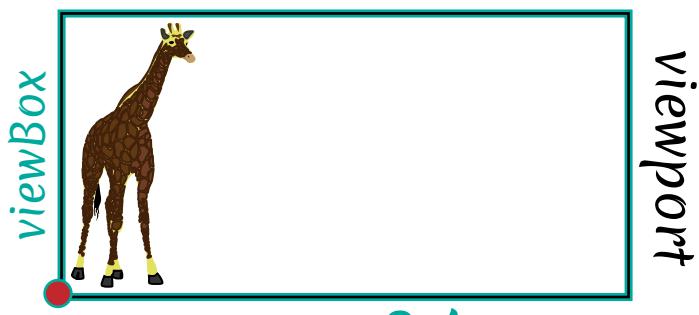
viewport



3.2



3.3



3.4

viewport

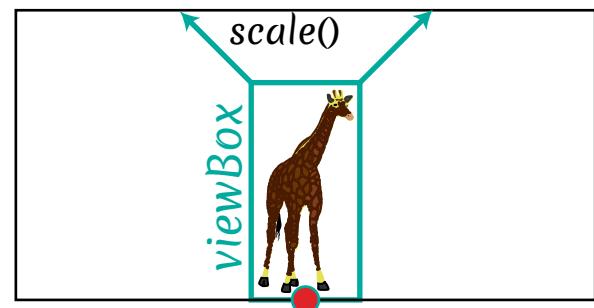
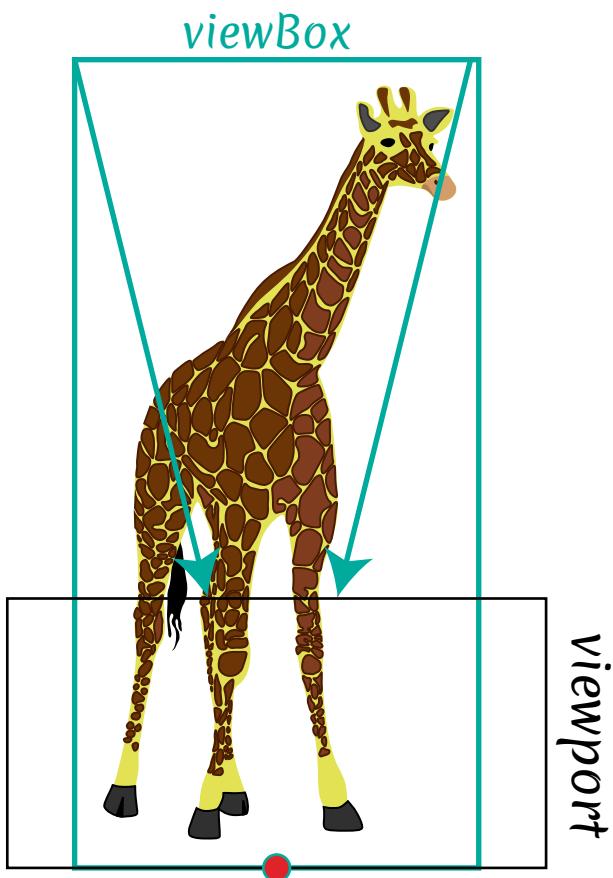
viewport

viewport

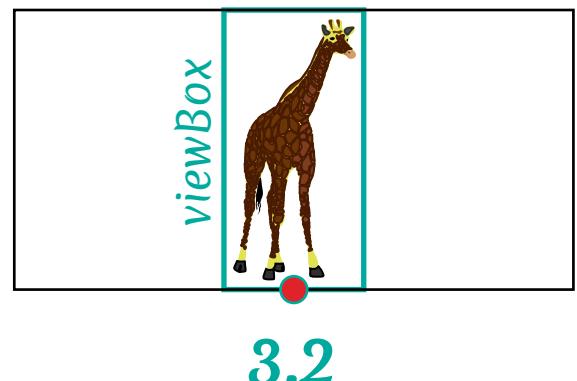
viewport

lesson 4

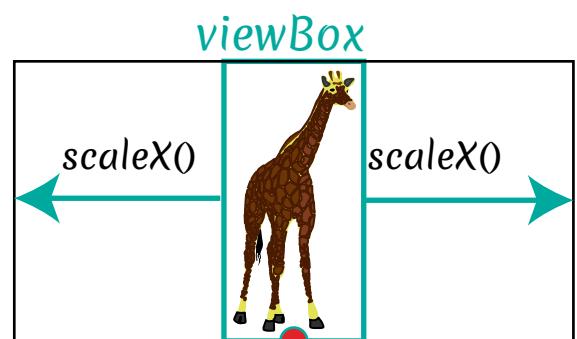
PRESERVEASPECTRATIO



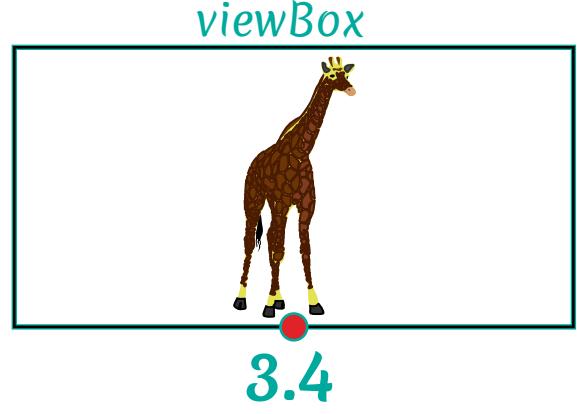
3.1



3.2



3.3

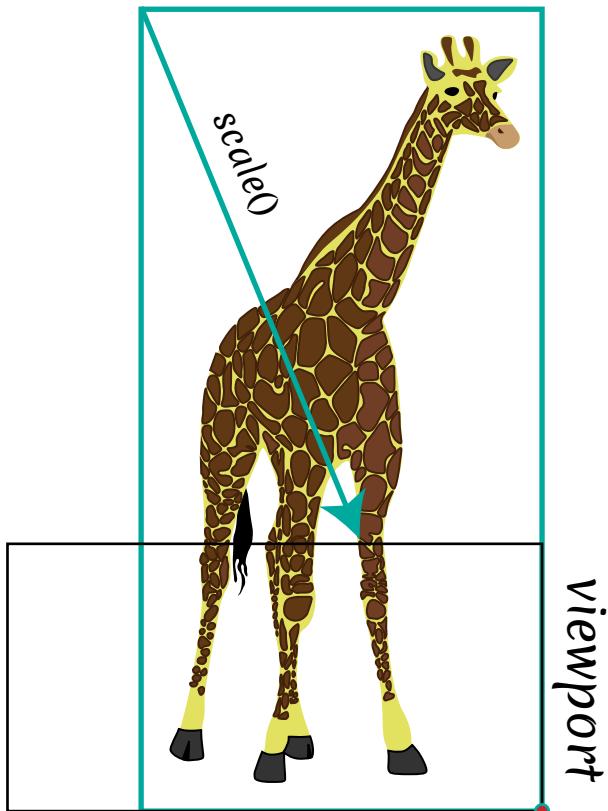


3.4

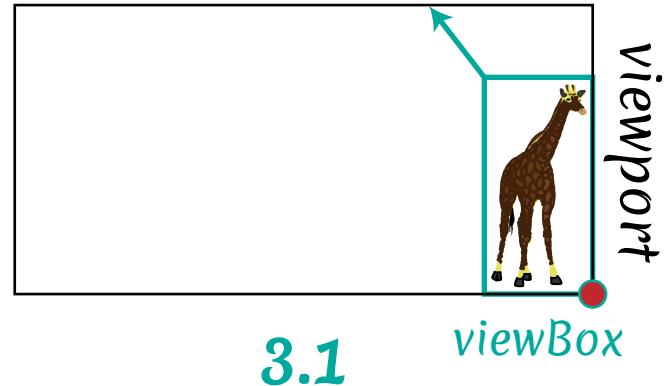
lesson 4

PRESERVEASPECTRATIO

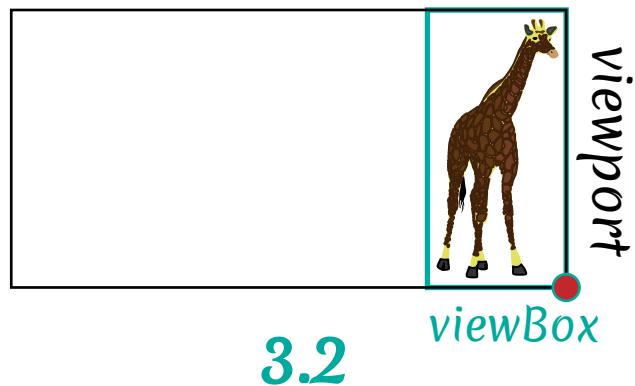
viewBox



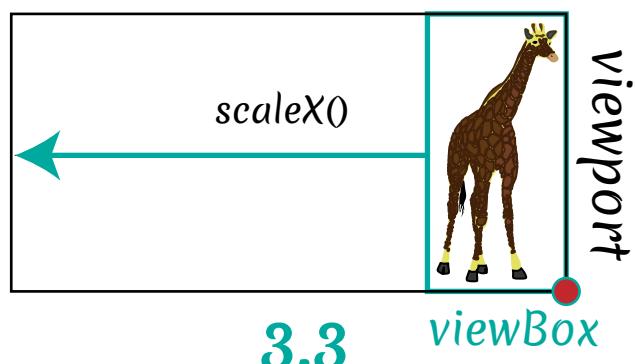
3.1



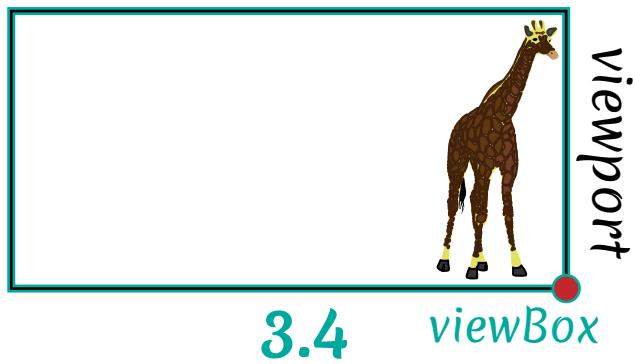
3.1



3.2



3.3



3.4

lesson 4

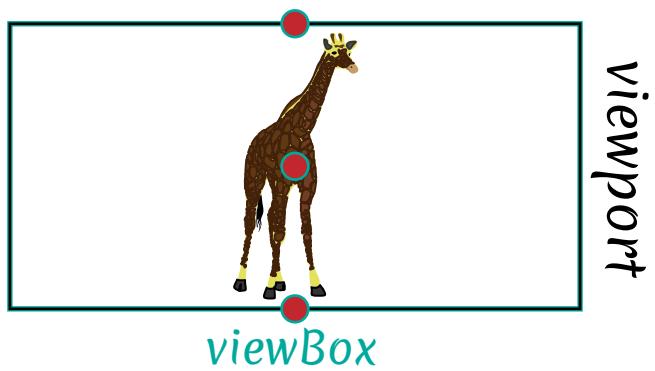
PRESERVEASPECTRATIO

meet

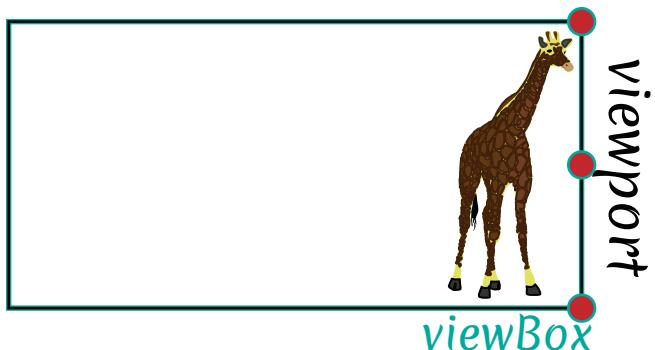


<align>

xMinYMin
xMinYMid
xMinYmax



xMidYMin
xMidYMid
xMidYmax



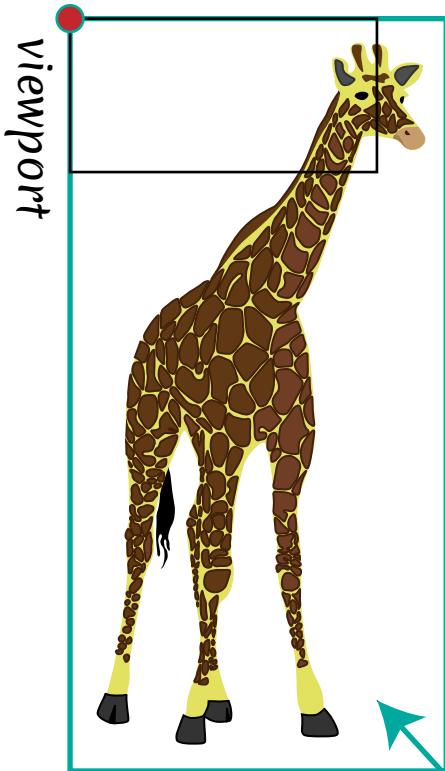
xMaxYMin
xMaxYMid
xMaxYMax

As you may have noticed, If the viewport is horizontal and the viewBox is vertical, then the alignment is horizontal. So it doesn't matter what xM** value we use. The result will be the same.

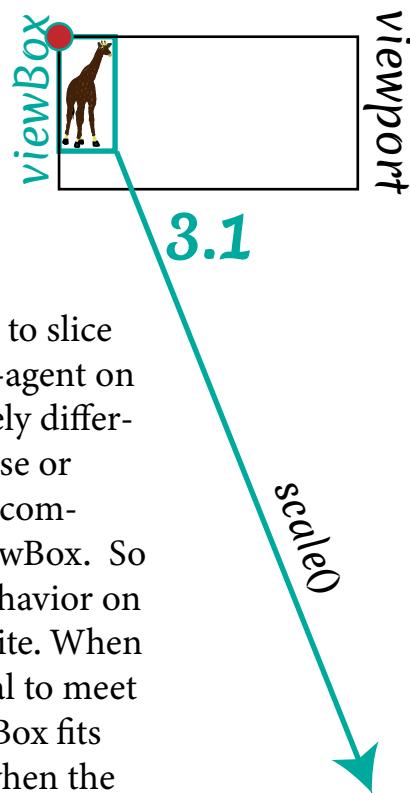
lesson 4

PRESERVEASPECTRATIO

xMinYMin



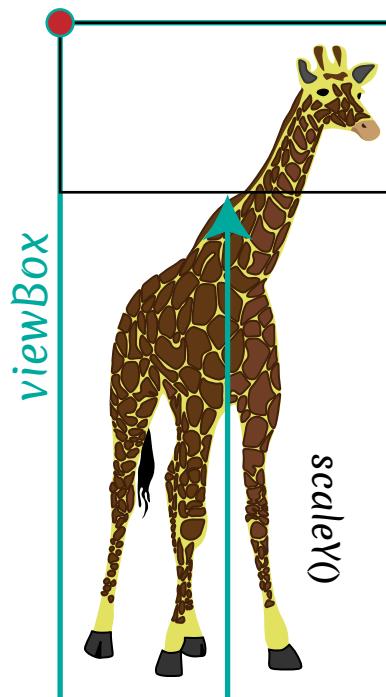
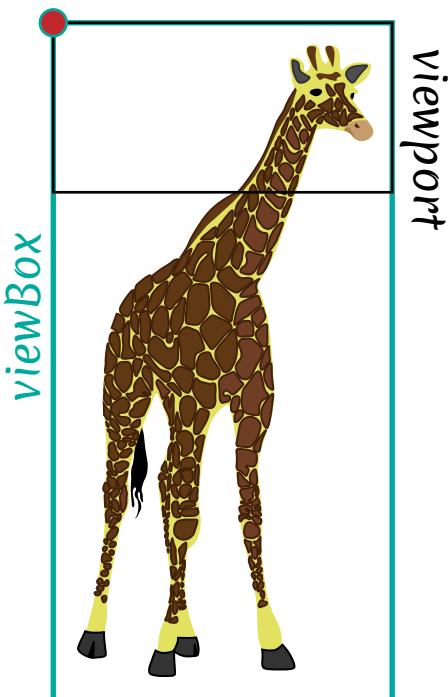
slice



OR

3.1

If the `meetOrSlice` equal to slice the behavior of the user-agent on the third step is absolutely different. The viewBox increase or decrease until viewport completely fits inside the viewBox. So we could say that the behavior on the step 3.1 is the opposite. When the `<meetOrSlice>` equal to meet on the step 3.1 the viewBox fits into the viewport, but when the `<meetOrSlice>` is set to slice on



3.2

3.3

3.4

the step 3.1 the viewport fits into the viewBox. On step 3.3 we can see that the size of the viewBox decrease to be equal to the size of the viewport by using the `scaleX` or `scaleY` functions. We could use the metaphor that we take a knife and slice the viewBox. The parts of the viewBox that lays outside the boundaries of the viewport become invisible.

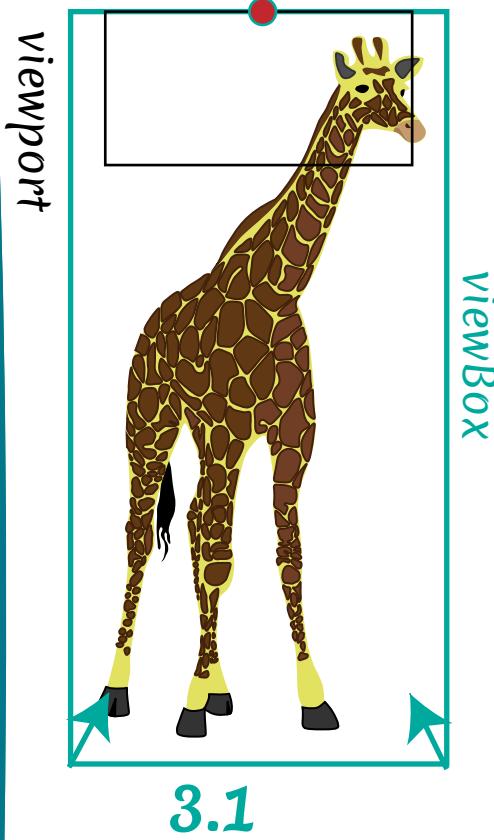
lesson 4

PRESERVEASPECTRATIO

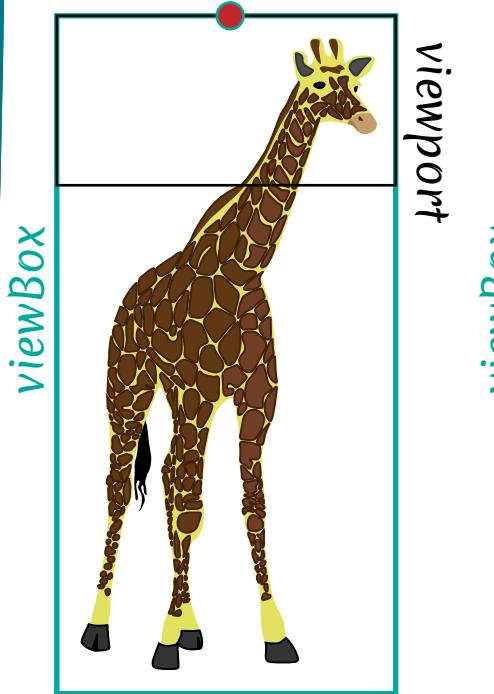
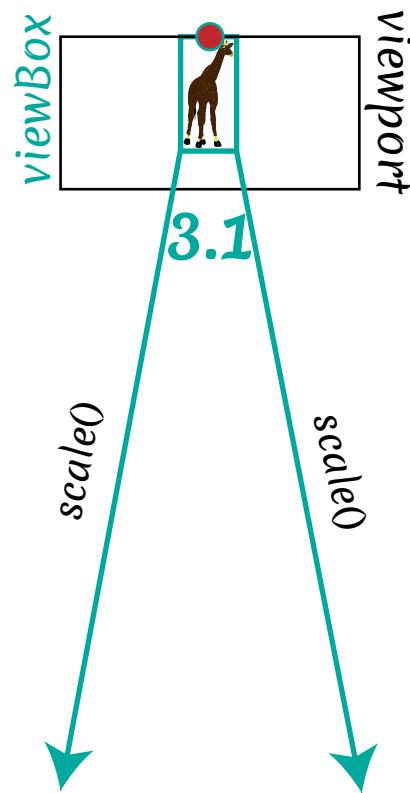
xMidYMin

slice

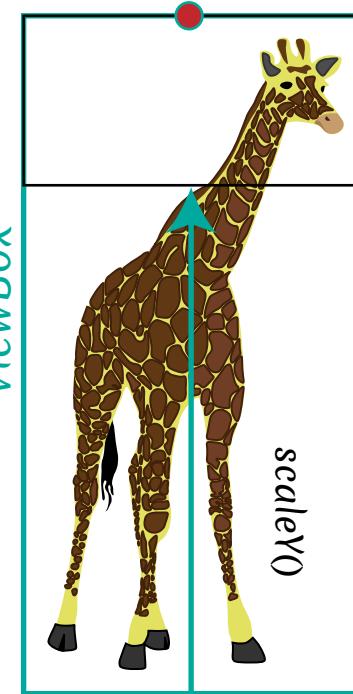
OR



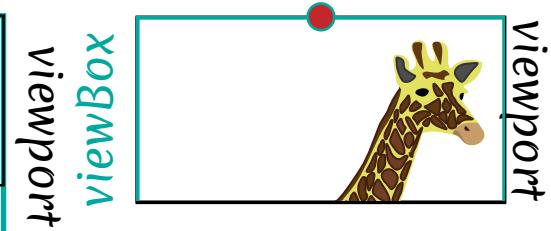
3.1



3.2



3.3

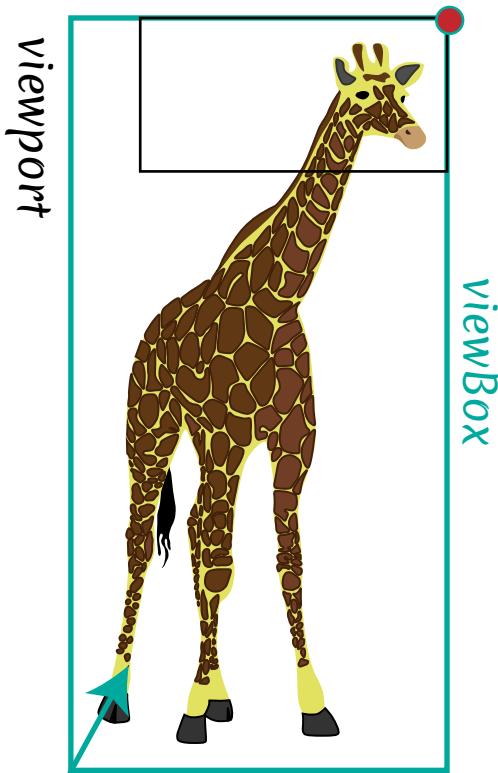


3.4

lesson 4

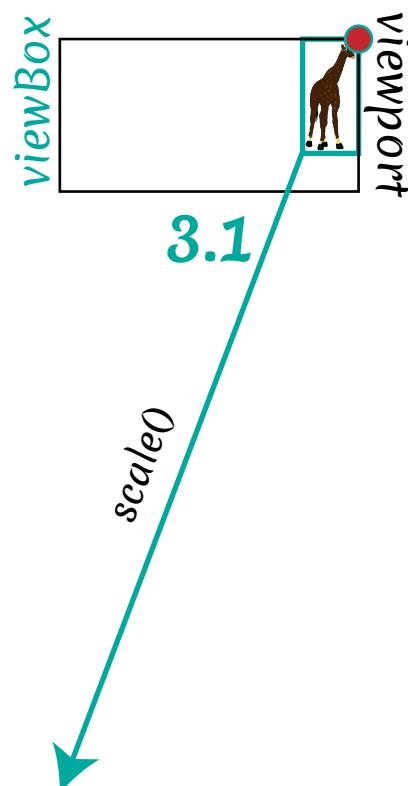
PRESERVEASPECTRATIO

xMaxYMin



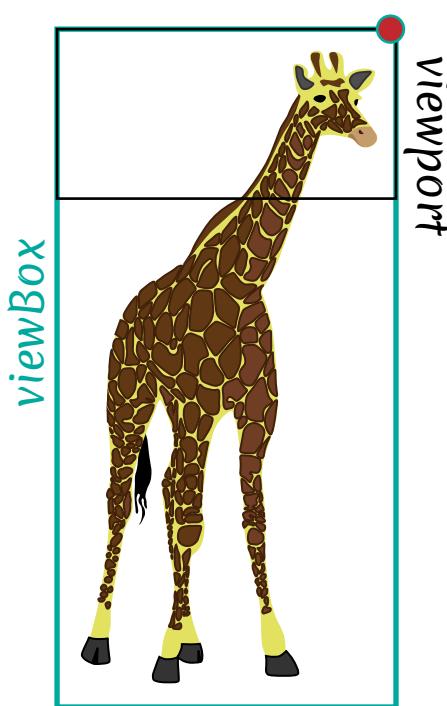
3.1

slice

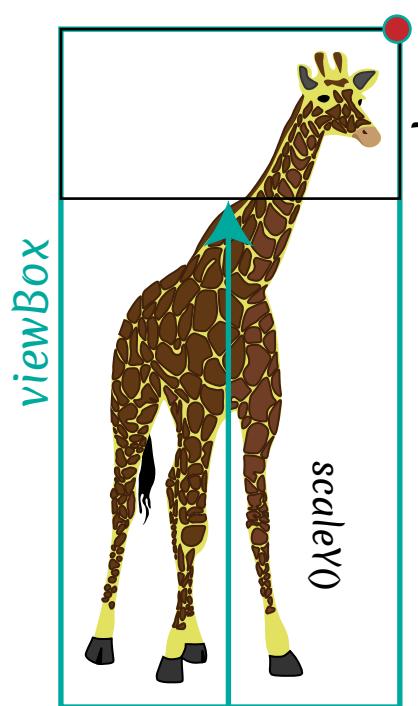


3.1

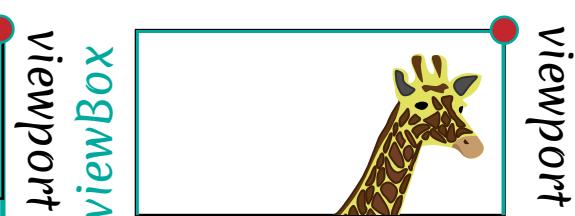
OR



3.2



3.3



3.4

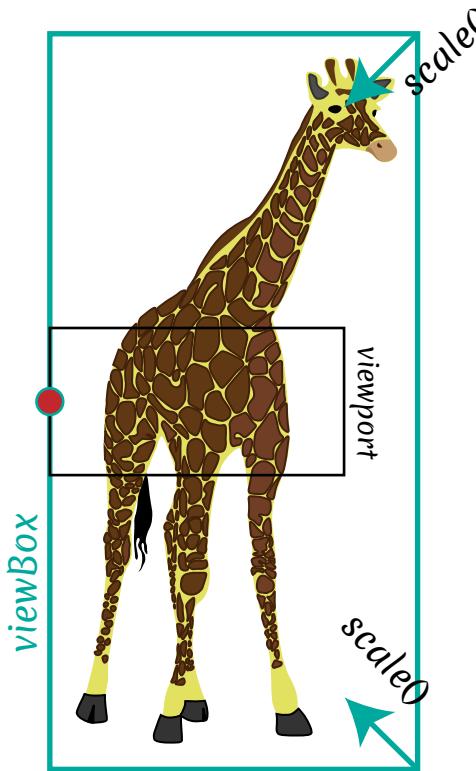
scaleY0

lesson 4

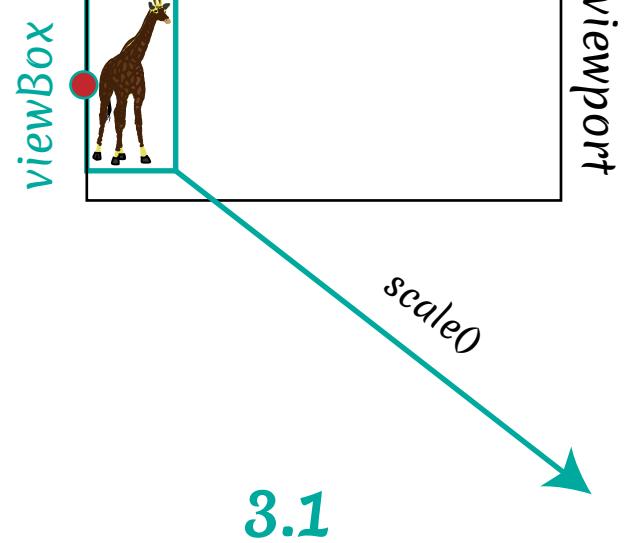
PRESERVEASPECTRATIO

xMinYMid

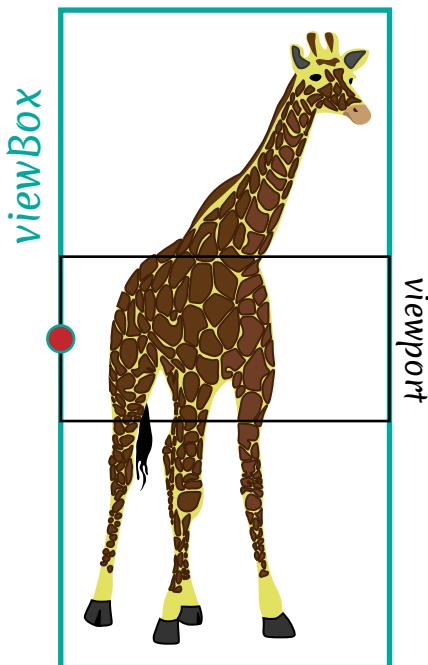
slice



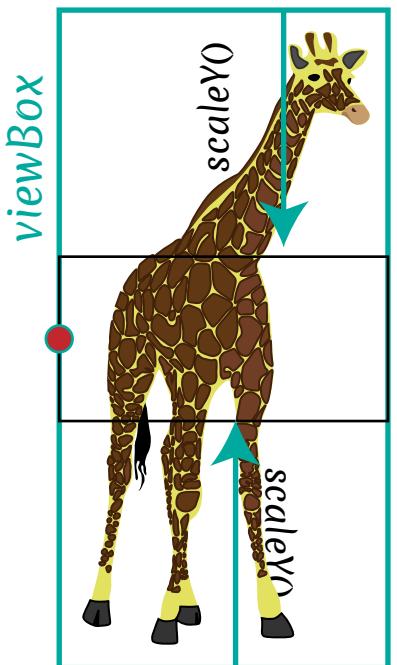
3.1



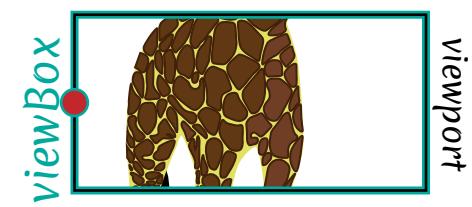
3.1



3.2



3.3



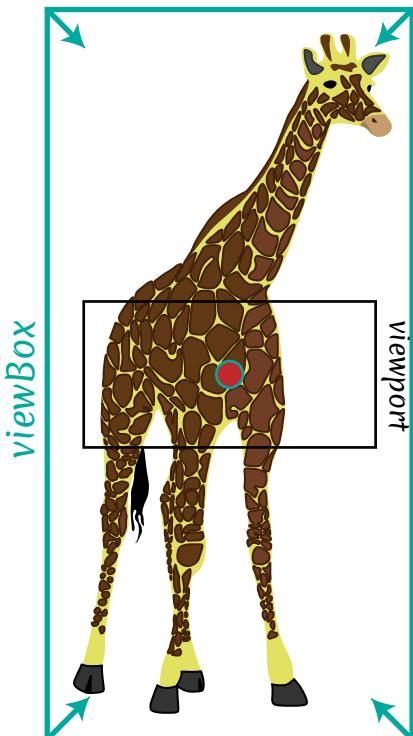
3.4

lesson 4

PRESERVEASPECTRATIO

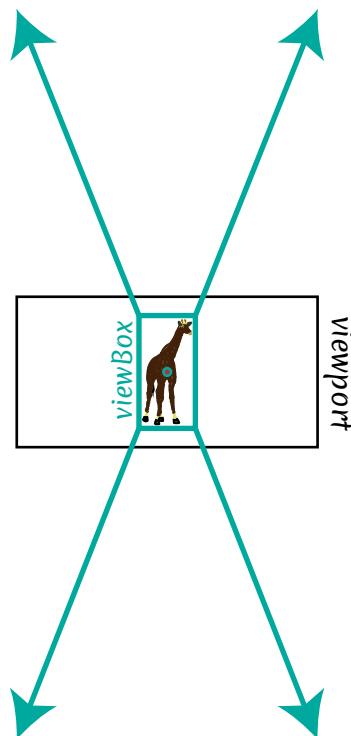
xMidYMid

slice

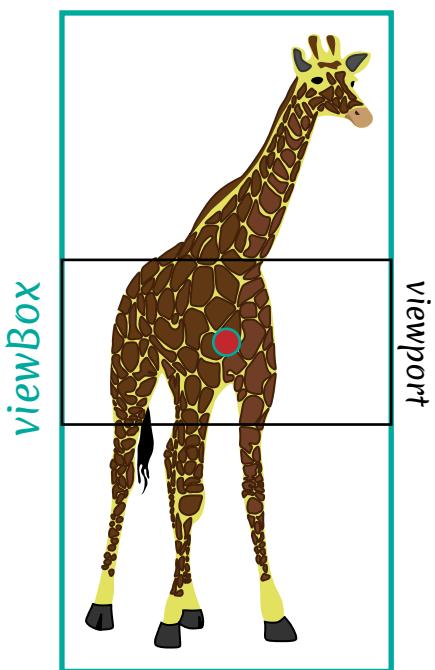


3.1

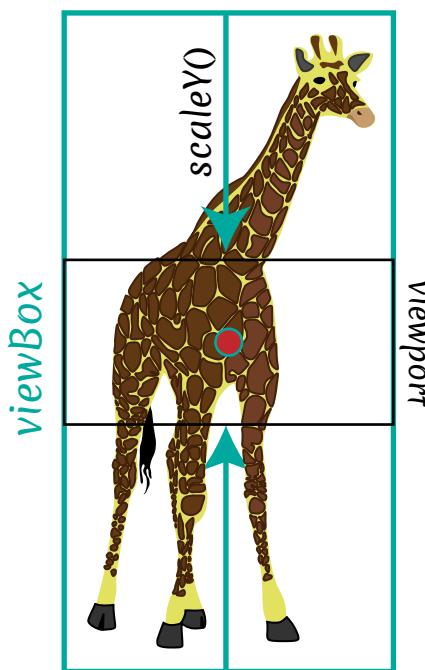
OR



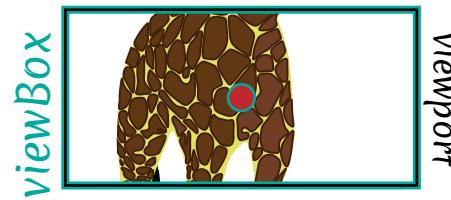
3.1



3.2



3.3



3.4

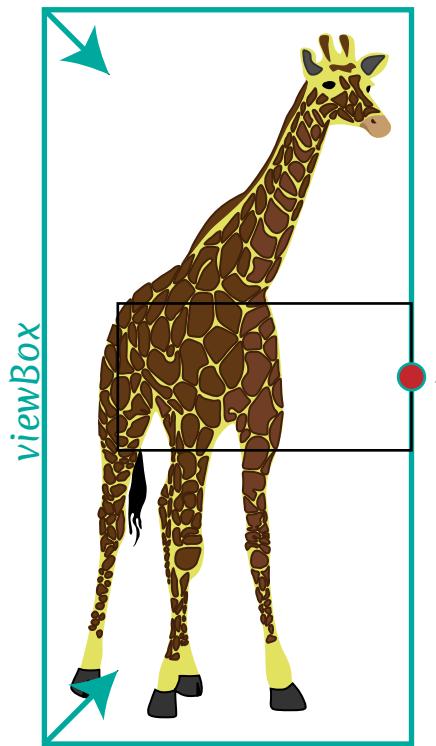
viewport

lesson 4

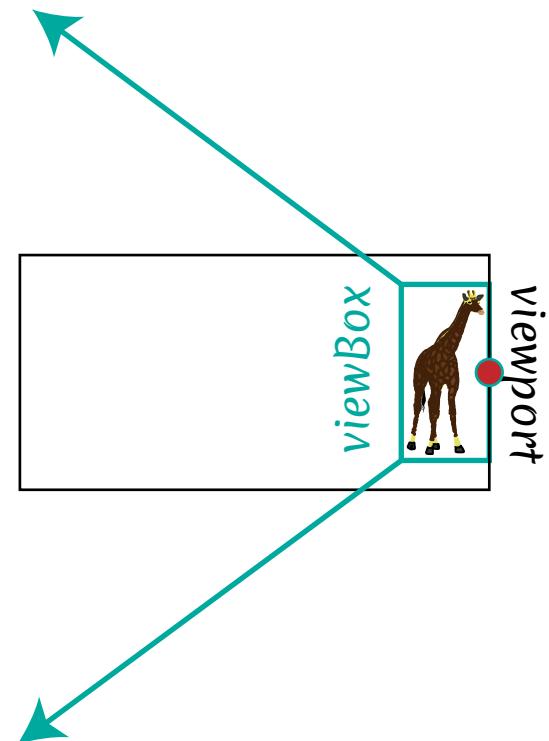
PRESERVEASPECTRATIO

xMaxYMid

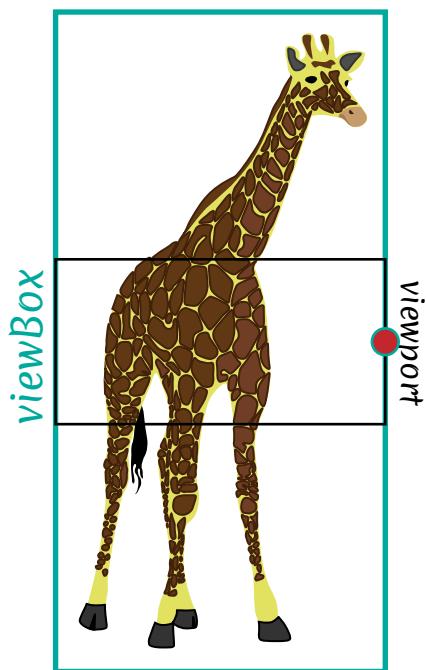
slice



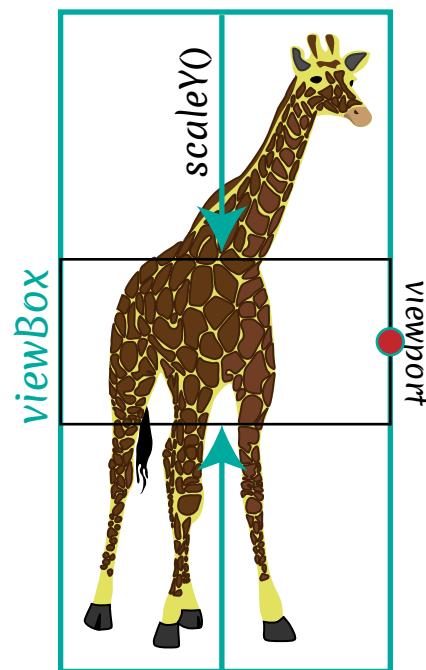
OR



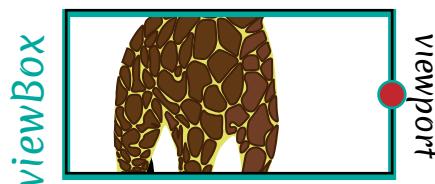
3.1



3.2



3.3



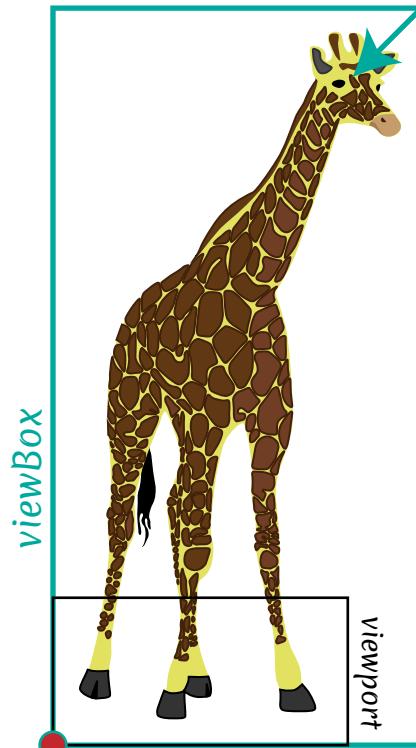
3.4

lesson 4

PRESERVEASPECTRATIO

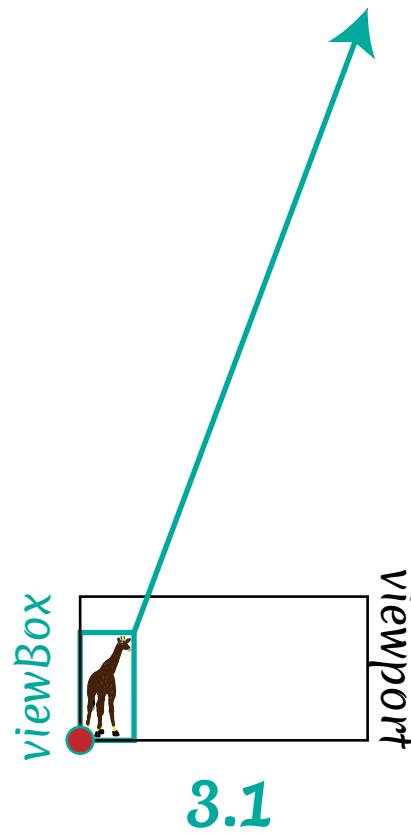
xMinYMax

slice

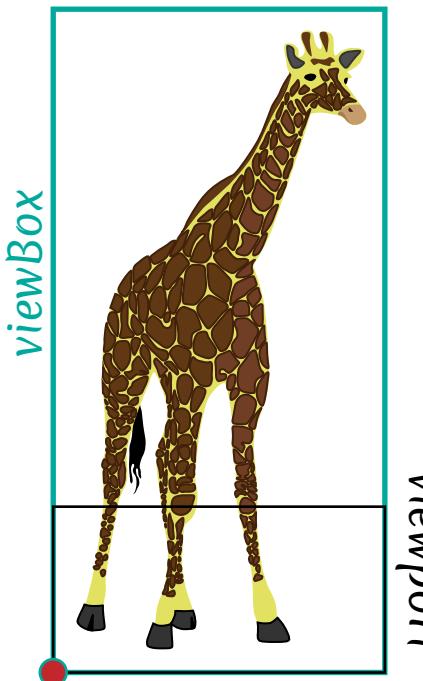


3.1

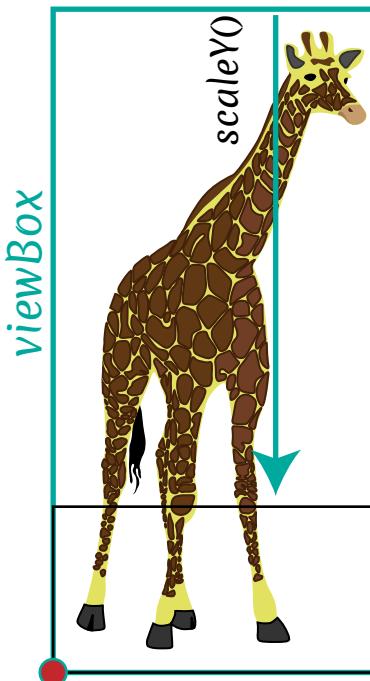
OR



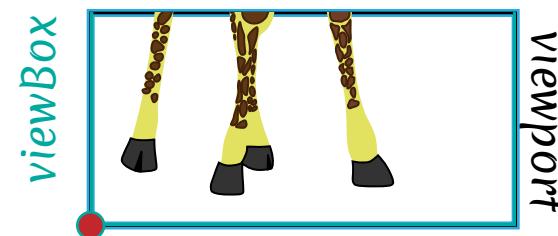
3.1



3.2



3.3

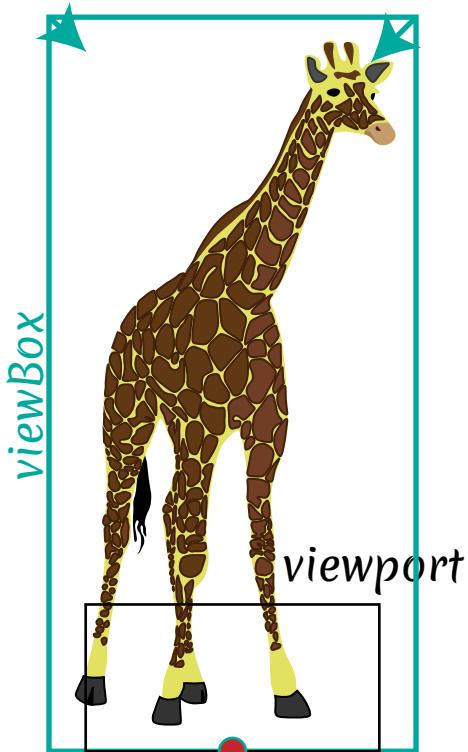


3.4

lesson 4

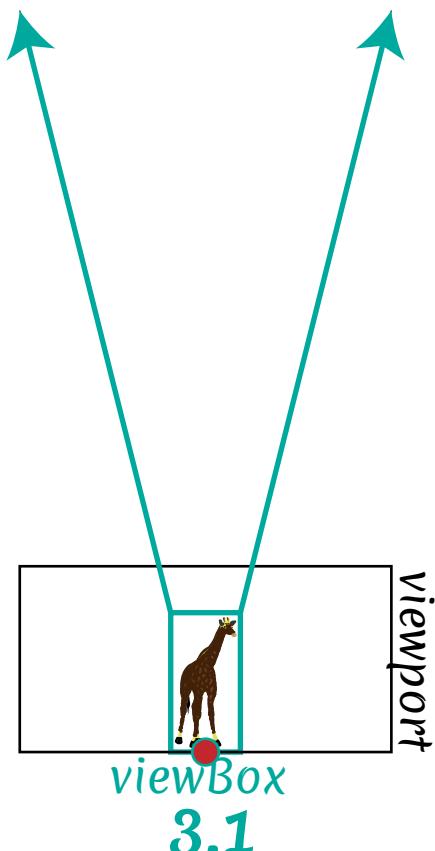
PRESERVEASPECTRATIO

xMidYMax



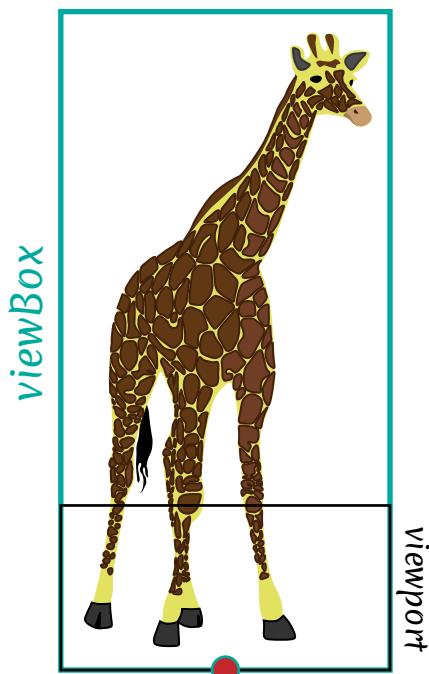
3.1

slice

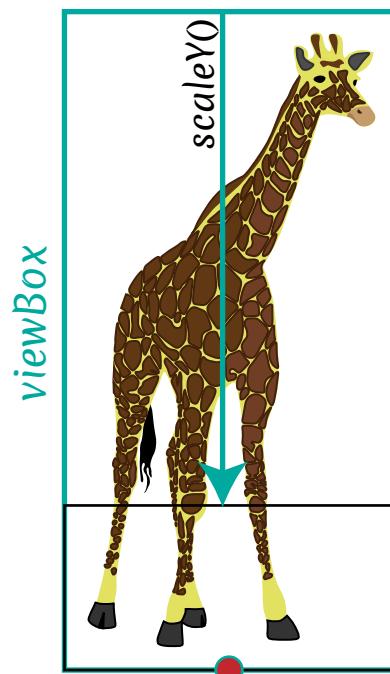


3.1

OR



3.2



3.3

viewBox



3.4

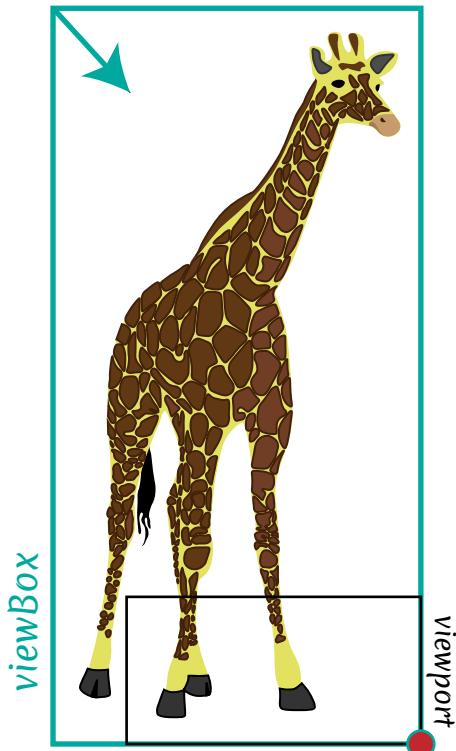
viewport

lesson 4

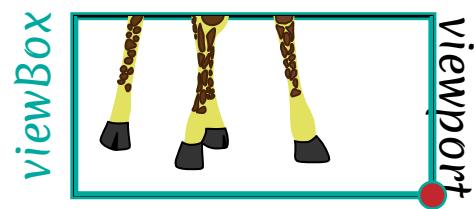
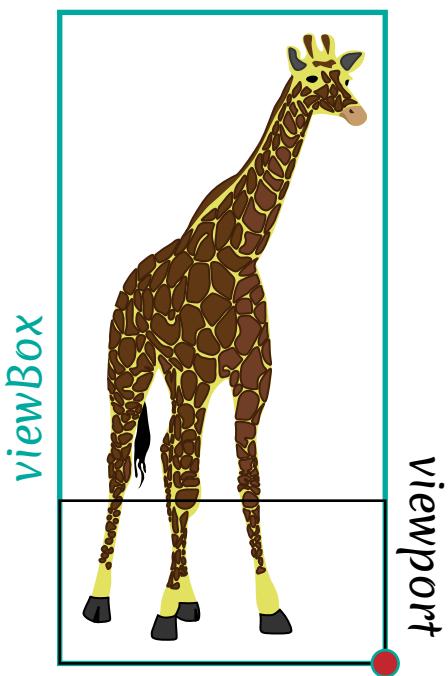
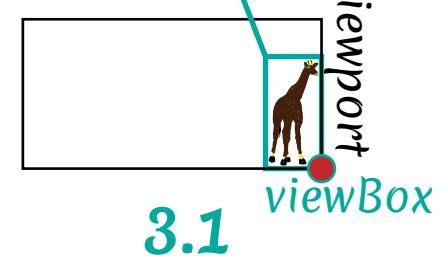
PRESERVEASPECTRATIO

xMaxYMax

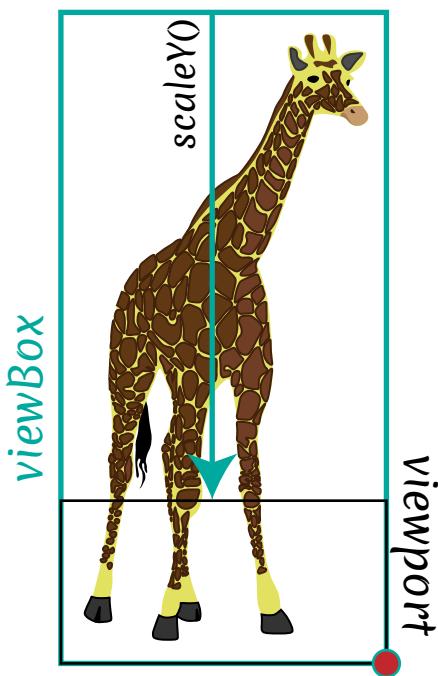
slice



3.1



3.2



3.3

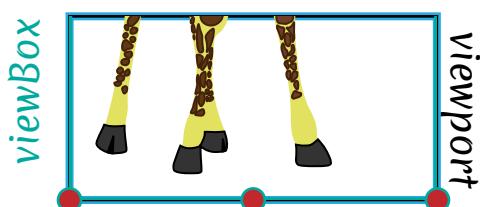
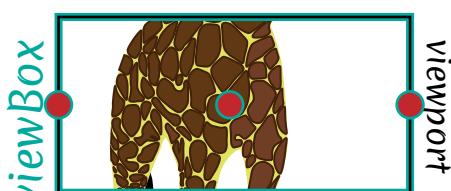
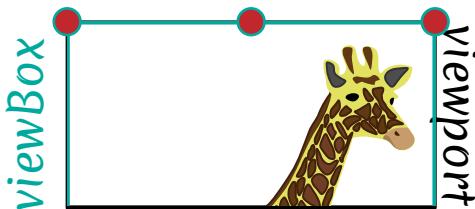
3.4

OR

lesson 4

PRESERVEASPECTRATIO

slice

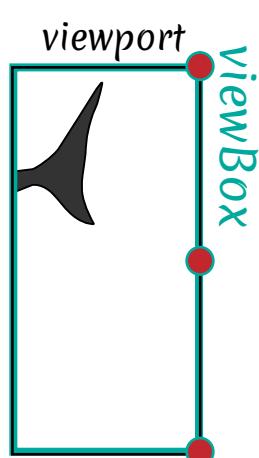
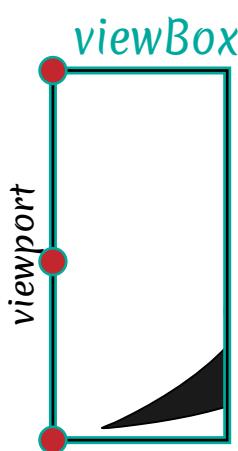


<align>

*xMinYMin
xMidYMin
xMaxYMin*

*xMinYMid
xMidYMid
xMaxYMid*

*xMinYMax
xMidYMax
xMaxYMax*



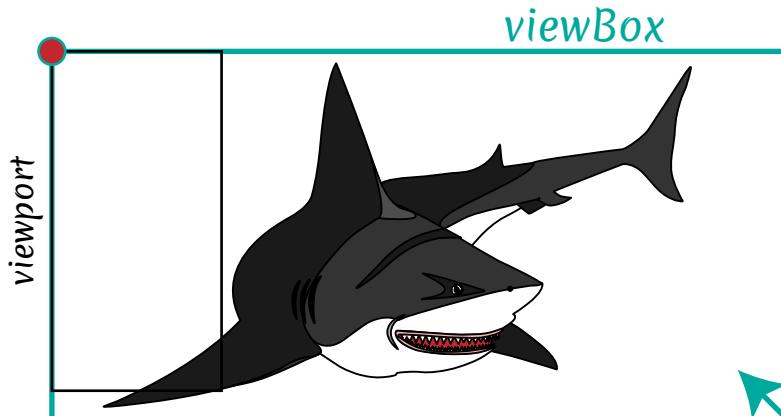
*xMinYMin
xMinYMid
xMinYMax*

*xMidYMin
xMidYMid
xMidYMax*

*xMaxYMin
xMaxYMid
xMaxYMax*

lesson 4

PRESERVEASPECTRATIO



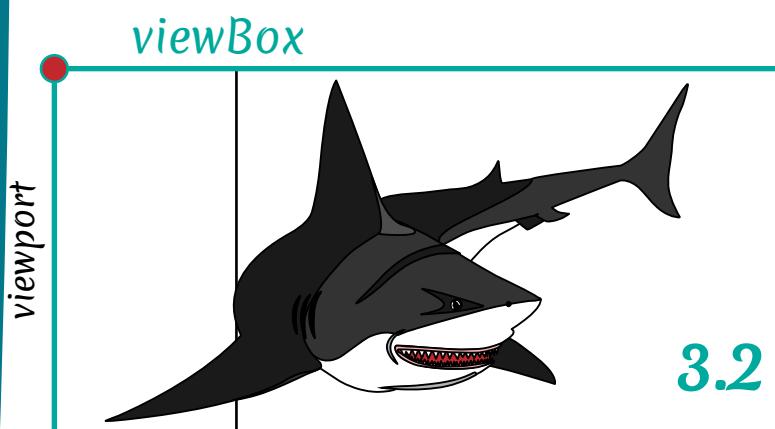
3.1

`xMinYMin`

`slice`

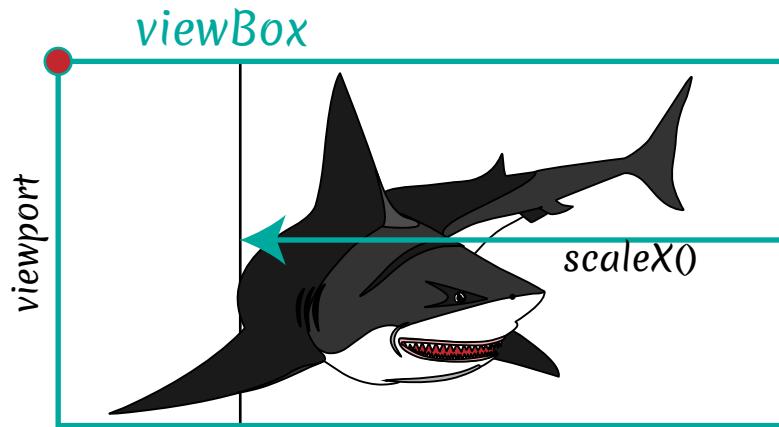
`viewBox`
OR

3.1

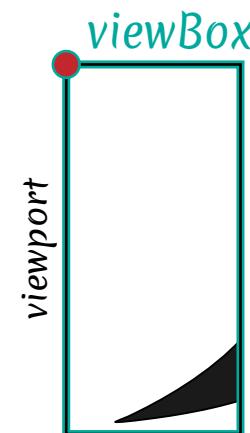


3.2

If the `meetOrSlice` equal to slice the behavior of the user-agent on the third step is absolutely different. The `viewBox` increase or decrease until `viewport` completely fits inside the `viewBox`. So we could say that the behavior on the step 3.1 is the opposite. When the `<meetOrSlice>` equal to meet on the step 3.1 the `viewBox` fits into the `viewport`, but when the `<meetOrSlice>` is set to slice on the step 3.1 the `viewport` fits into the `viewBox`. On step 3.3 we can see that the size of the `viewBox` decrease to be equal to the size of the `viewport` by using the `scaleX` or `scaleY` functions. We could use the metaphor that we take a knife and slice the `viewBox`. The parts of the `viewBox` that lays outside the boundaries of the `viewport` become invisible.



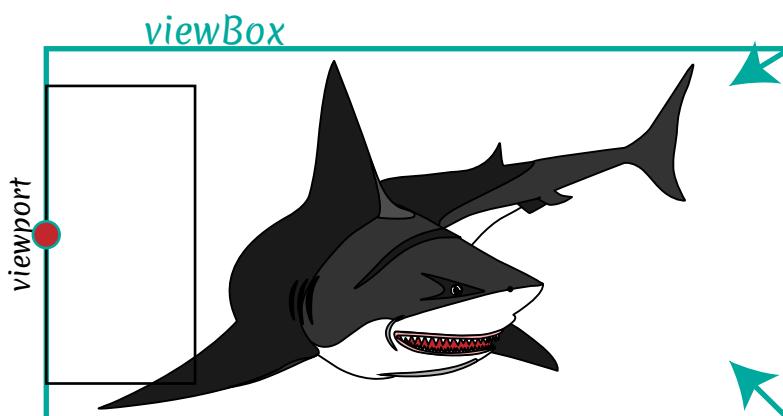
3.3



3.4

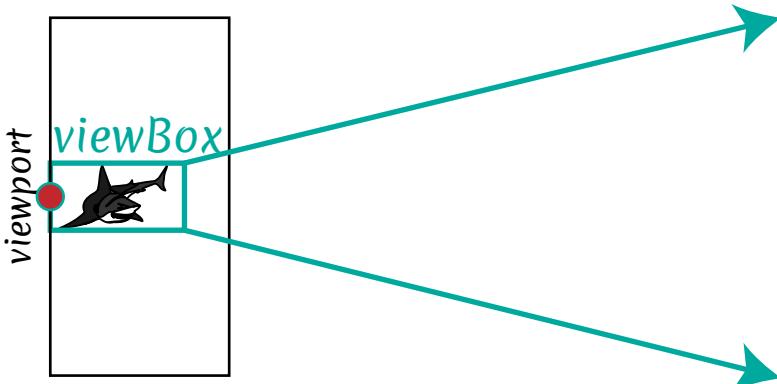
lesson 4

PRESERVEASPECTRATIO



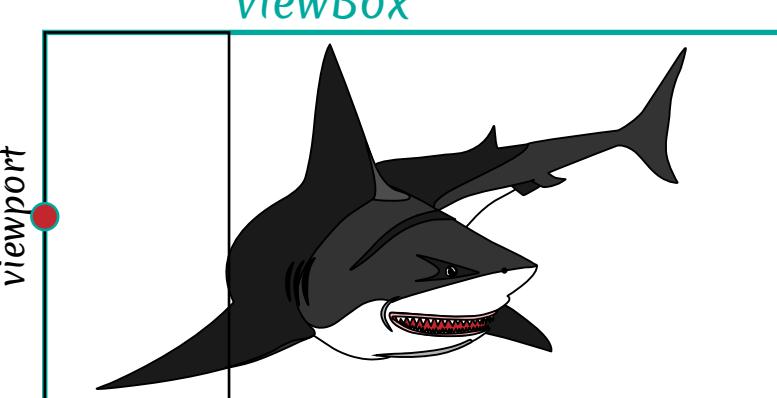
3.1

xMinYMid
slice

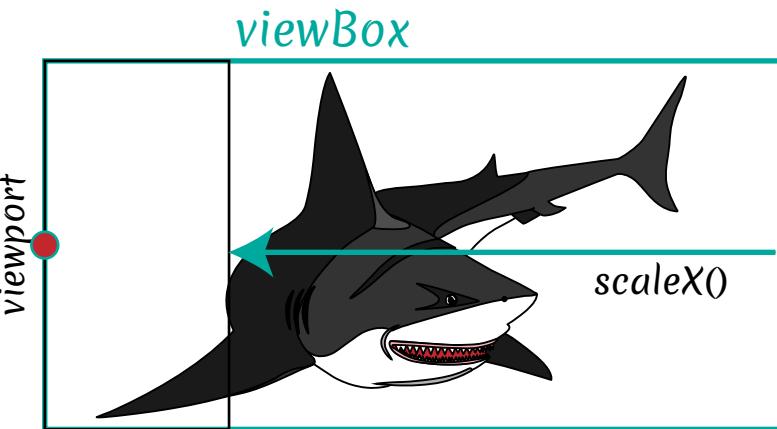


OR

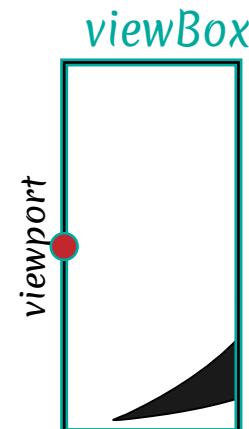
3.1



3.2



3.3

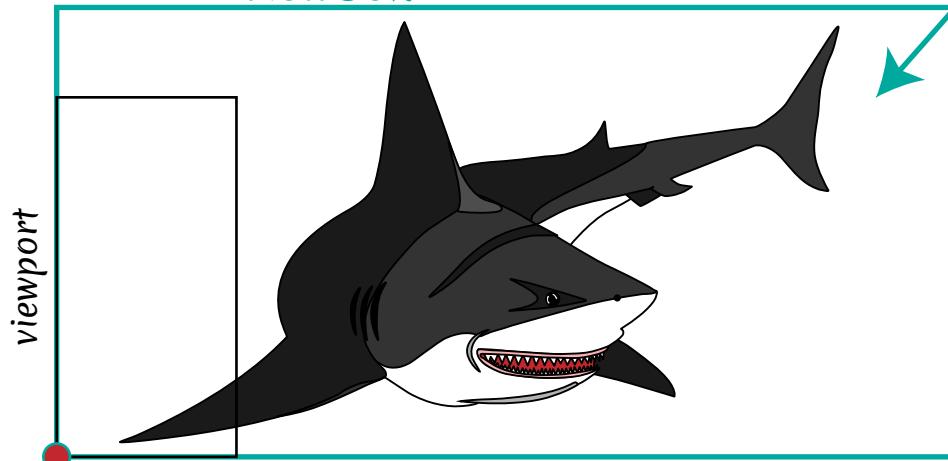


3.4

lesson 4

PRESERVEASPECTRATIO

viewBox

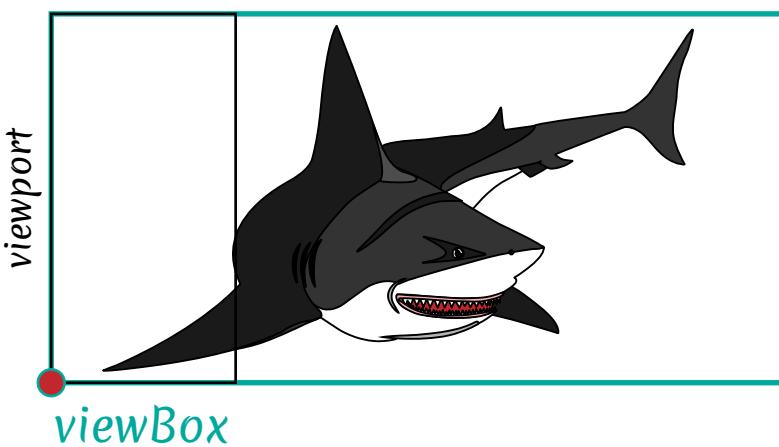
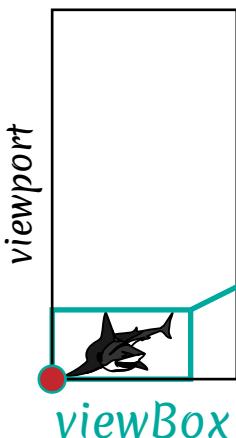


3.1

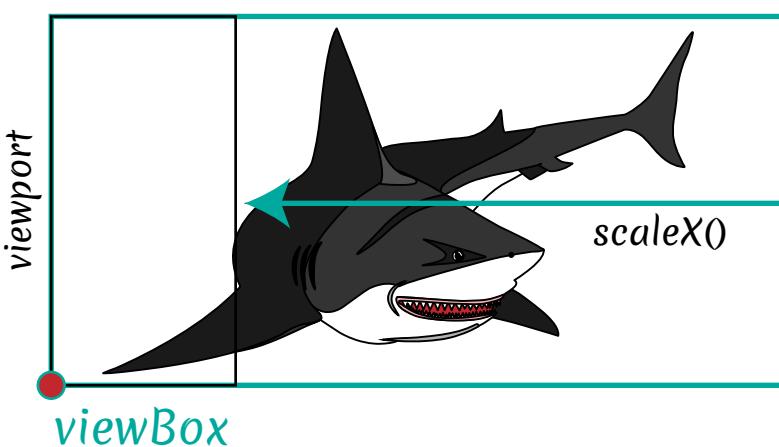
xMinYMax

OR

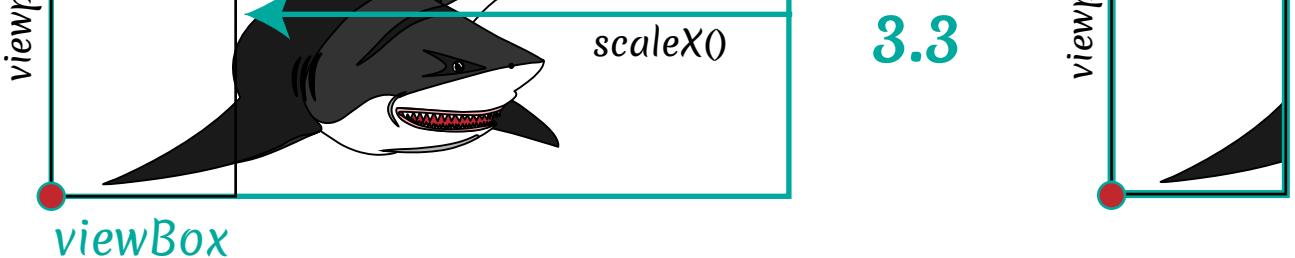
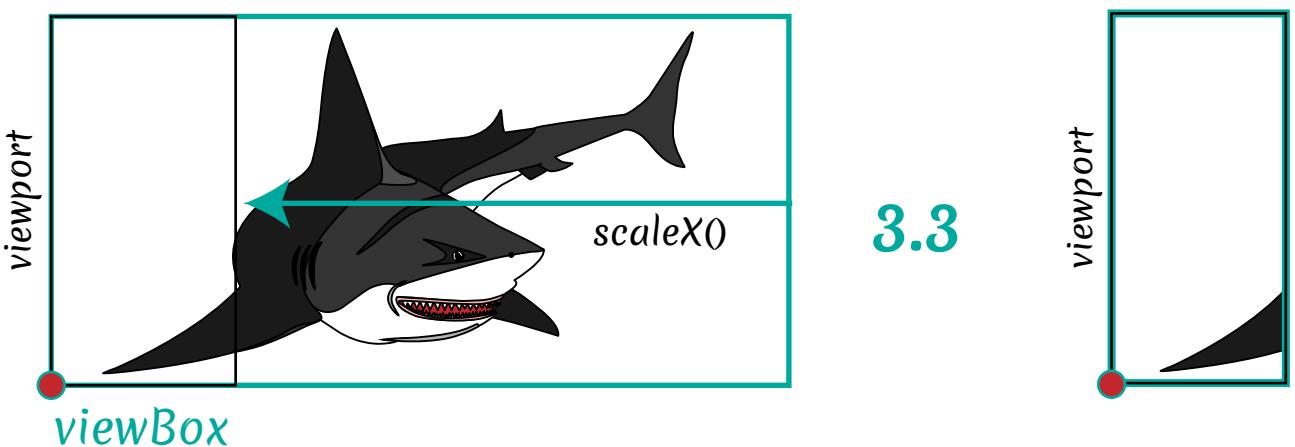
3.1



3.2



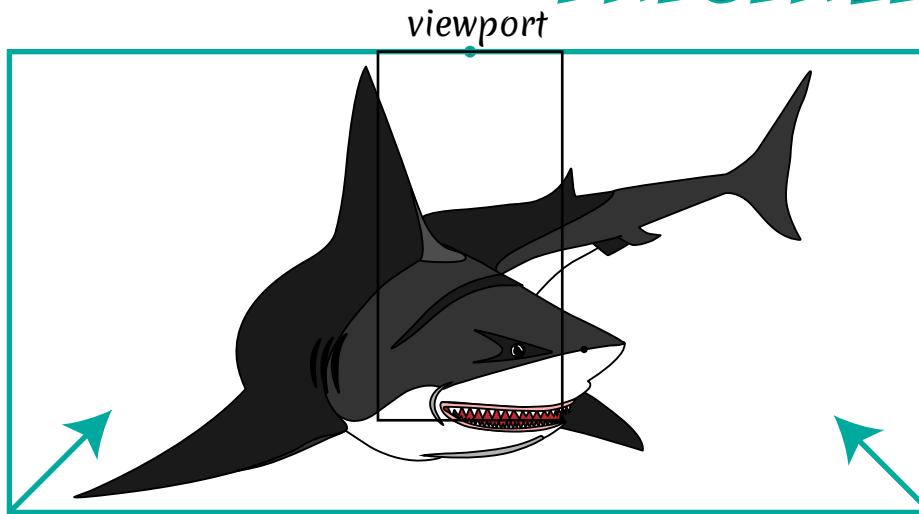
3.3



3.4

lesson 4

PRESERVEASPECTRATIO

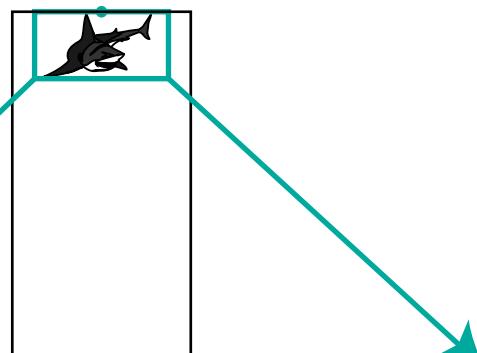


3.1

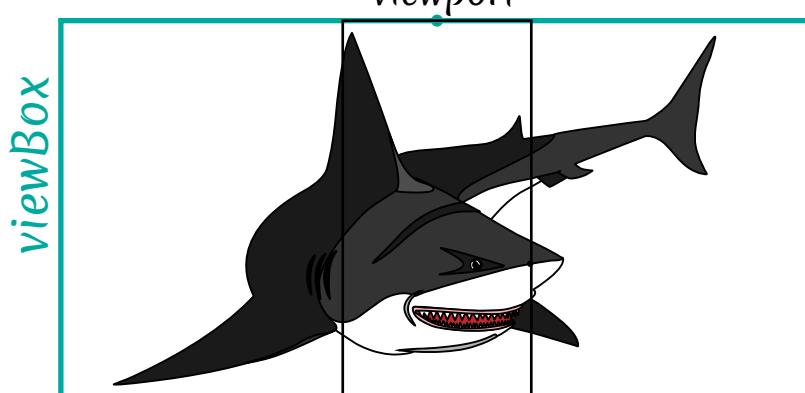
xMidYmin

slice

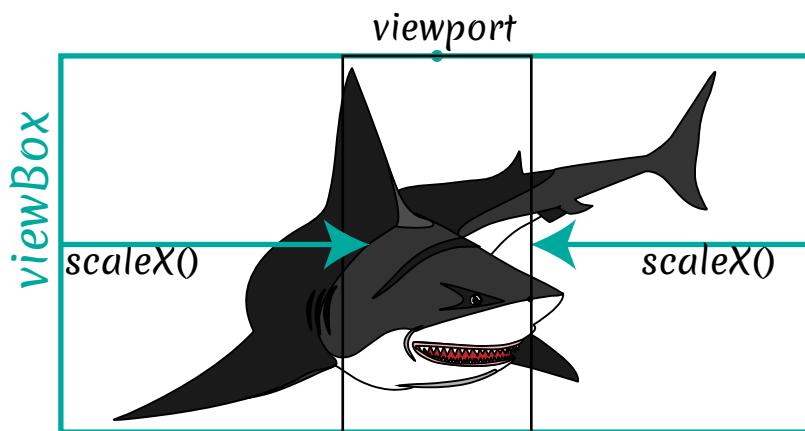
OR



3.1



3.2



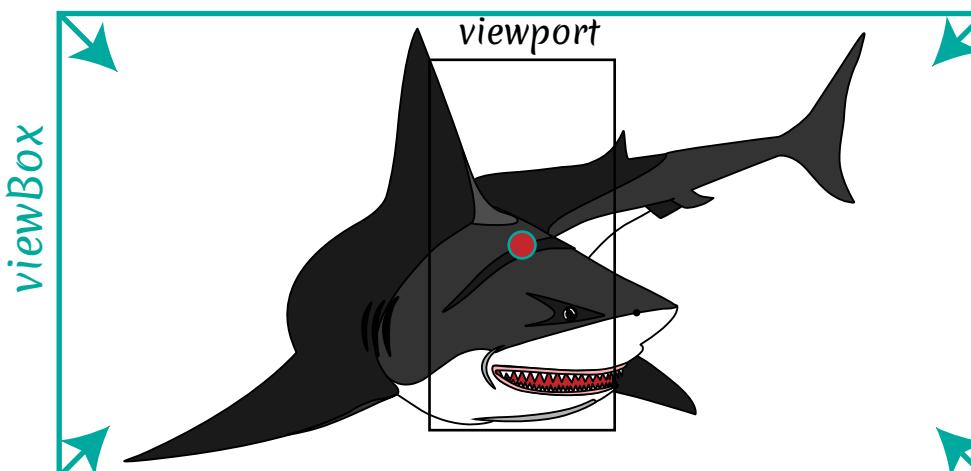
3.3



3.4

lesson 4

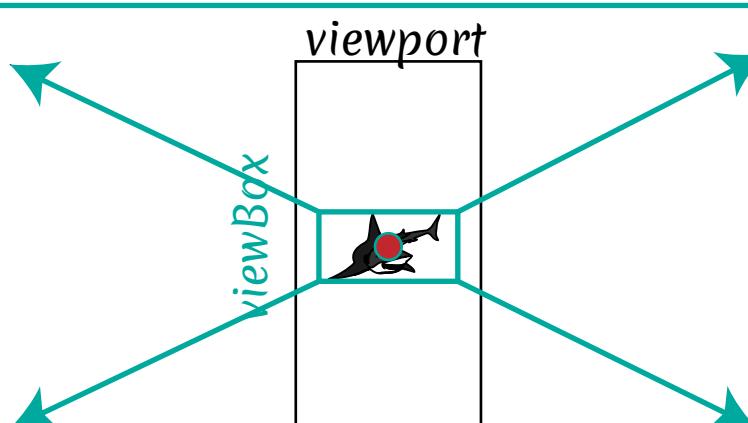
PRESERVEASPECTRATIO



`xMidYmid`

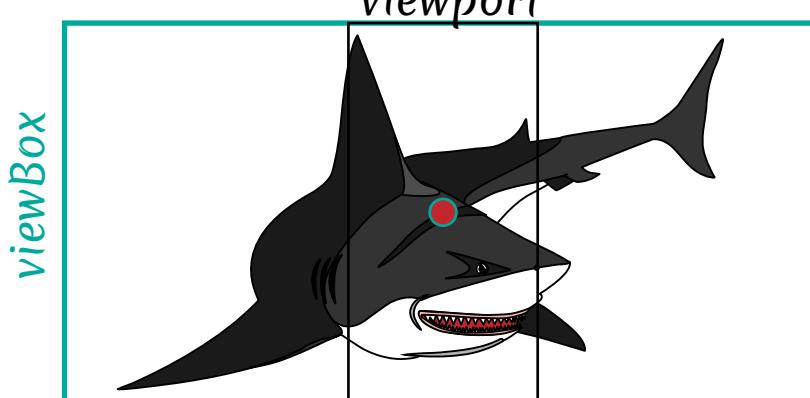
3.1

`slice`

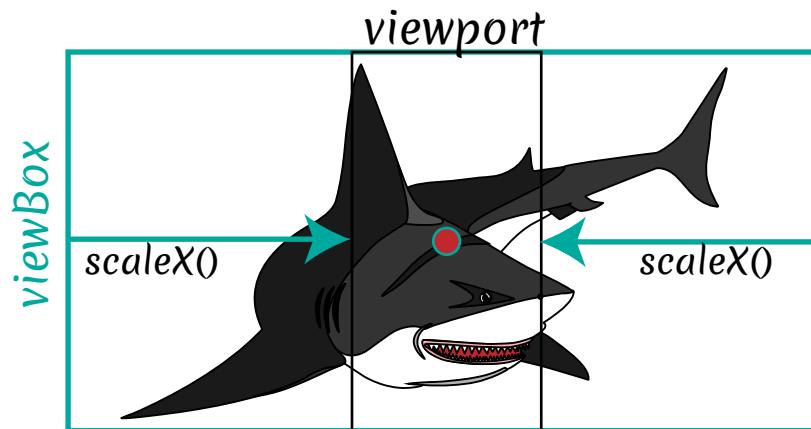


OR

3.1



3.2



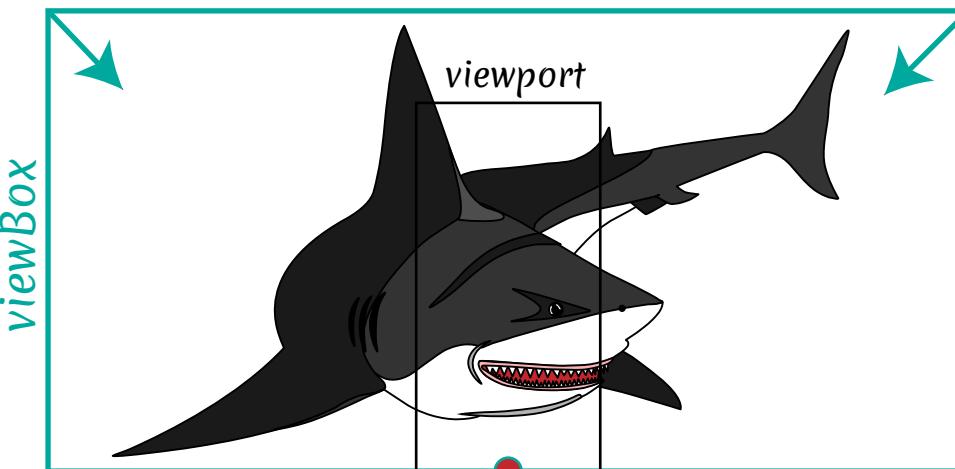
3.3



3.4

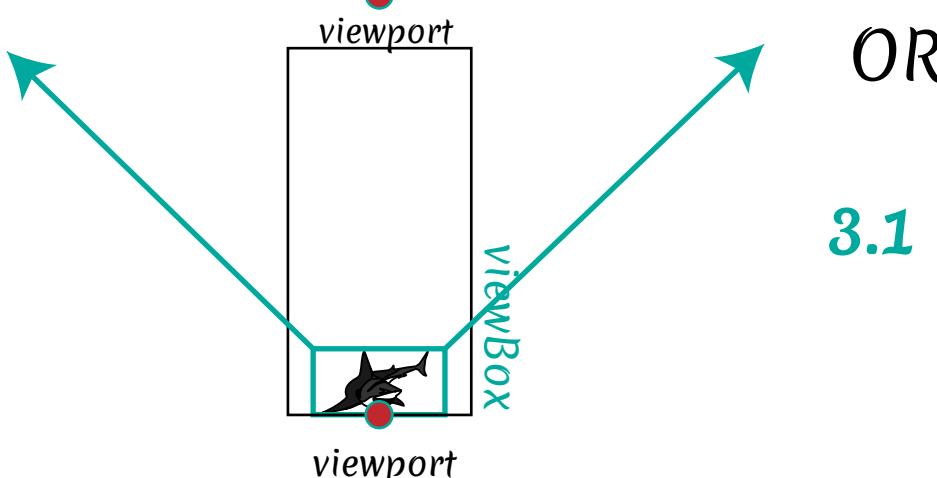
lesson 4

PRESERVEASPECTRATIO

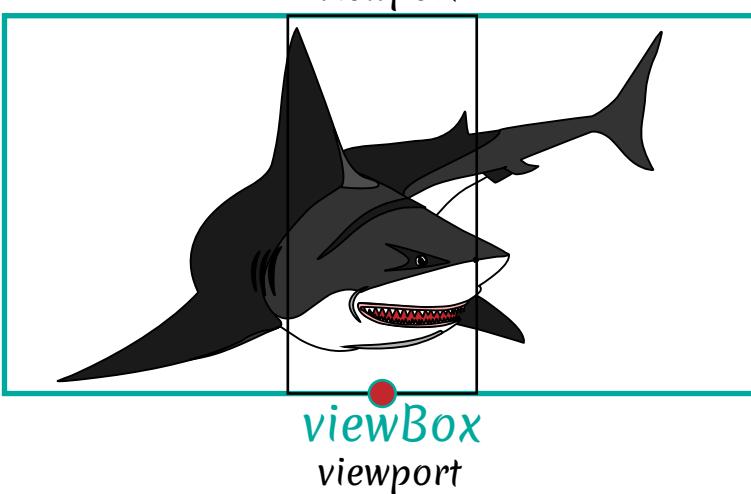


3.1

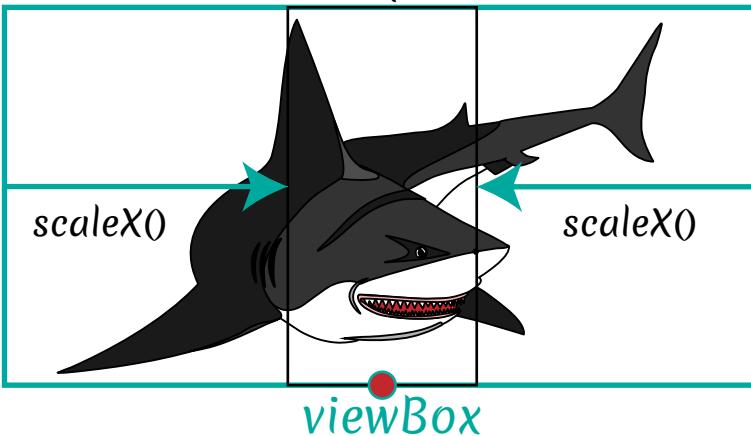
xMidYmax
slice



3.1



3.2



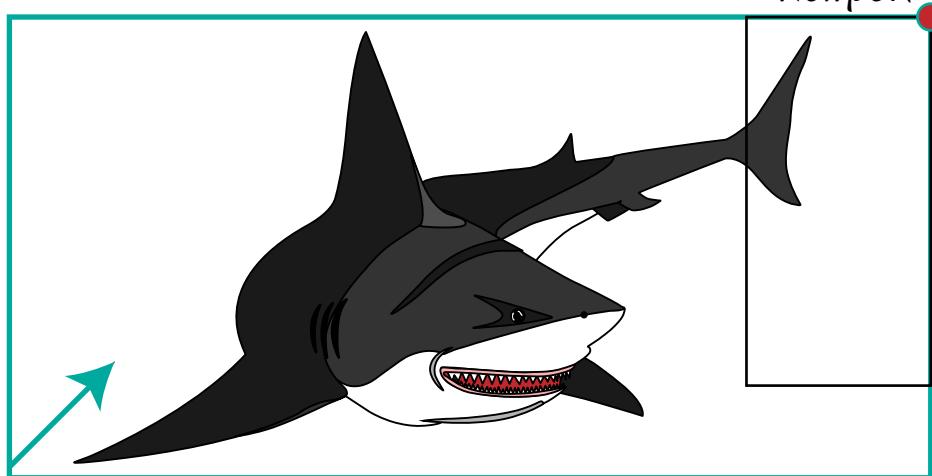
3.3



3.4

lesson 4

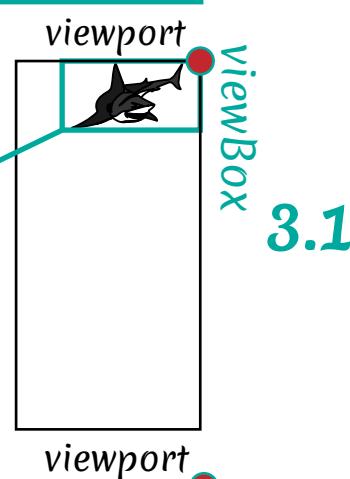
PRESERVEASPECTRATIO



`xMaxYmin`

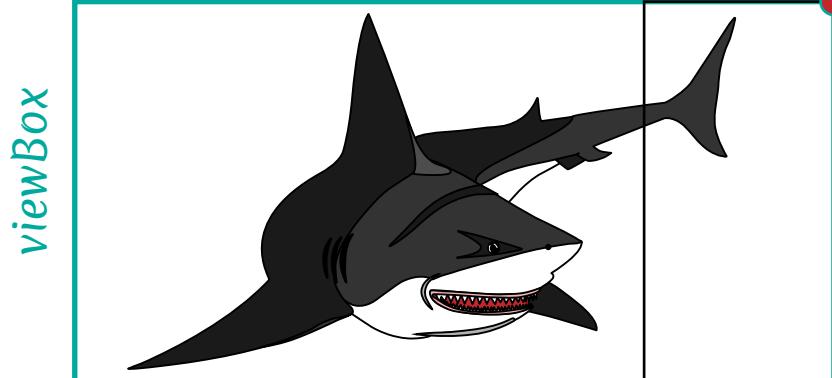
3.1

`slice`

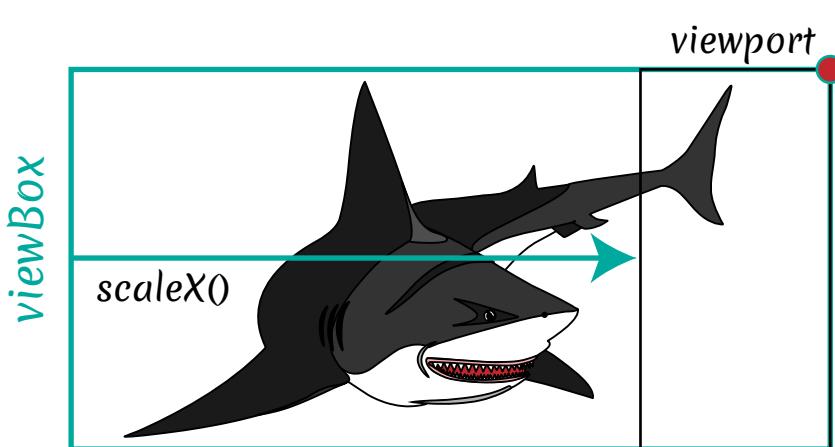


`viewport`

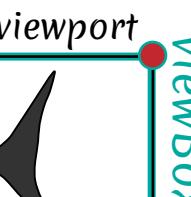
3.1



3.2



3.3

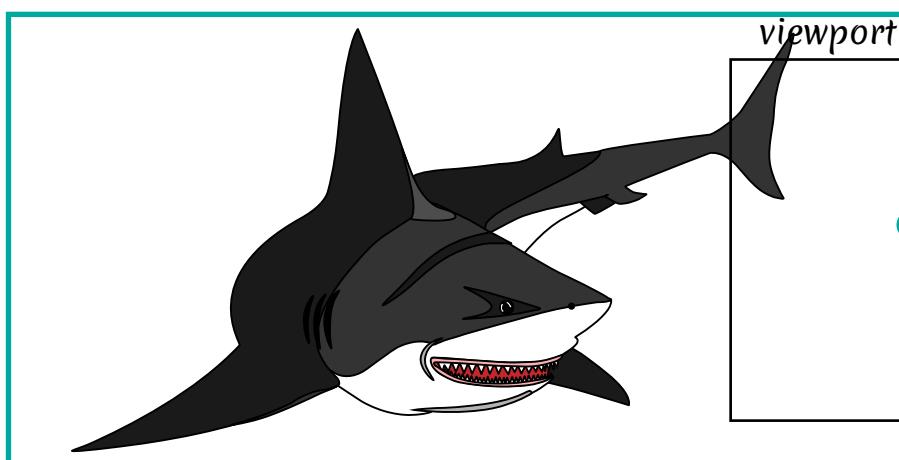


`viewport`

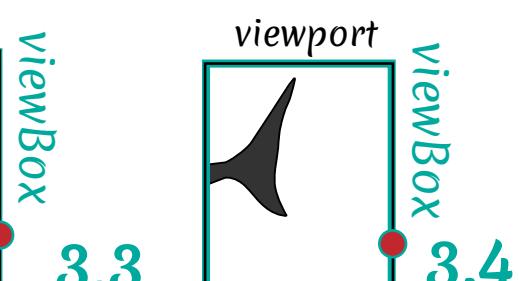
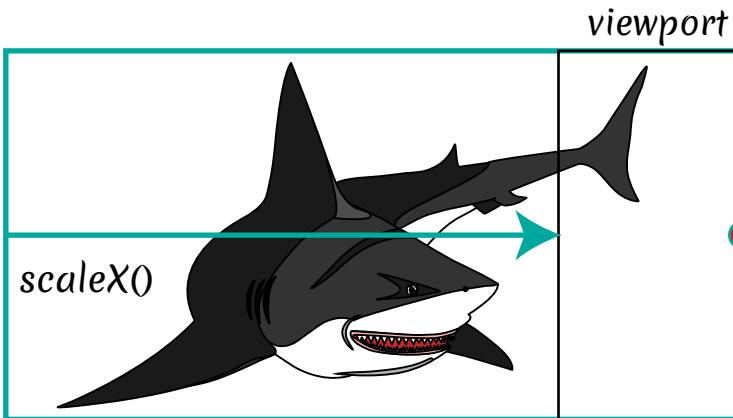
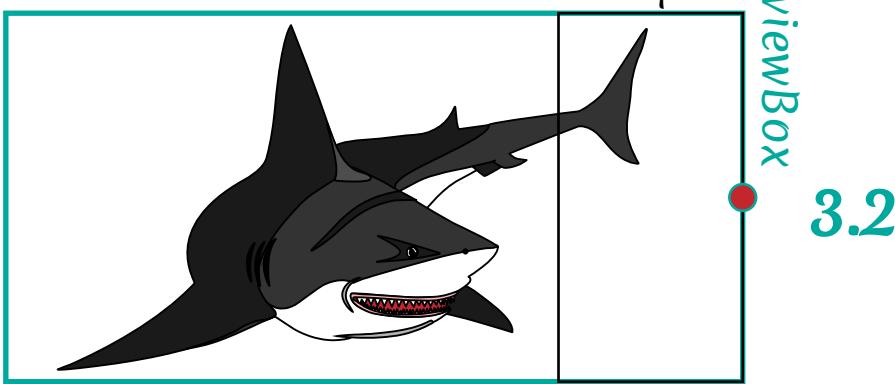
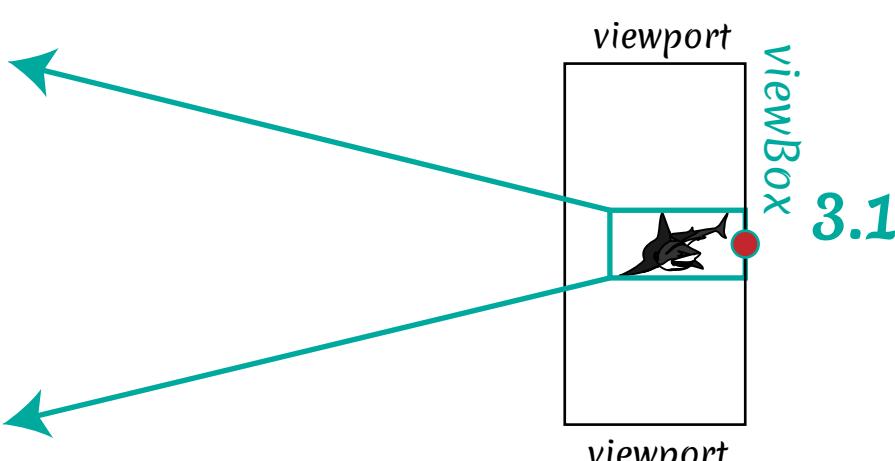
3.4

lesson 4

PRESERVEASPECTRATIO

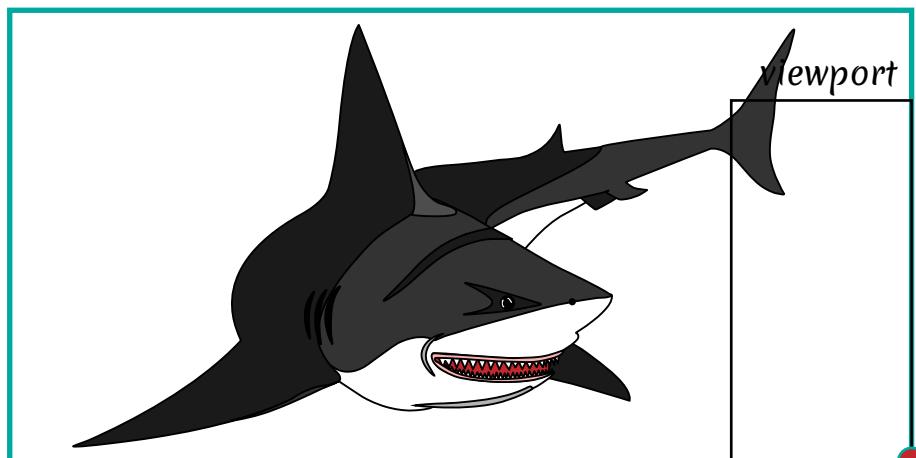


`xMaxYmid`
`slice`



lesson 4

PRESERVEASPECTRATIO



xMaxYmax
slice

