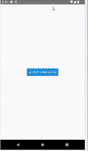
In this lesson we will learn about Activity Indicator(Loader) in React native whom we sometimes call loader in context of website or web application. You can call it activity indicator or Indicator or Loader it is upto you. All are same thing first of all Teacher will tell us that what are the use of activity indicator

0:16

You might have know it but again Teacher will slightly explain it so that you have an idea about it and then we will talk about common props here How to change its color or size and with a button how to hide and show it like by clicking on a button an activity indicator is shown and after sometime it will get hidden Now what is its use and why we do it Teacher will tell about it.

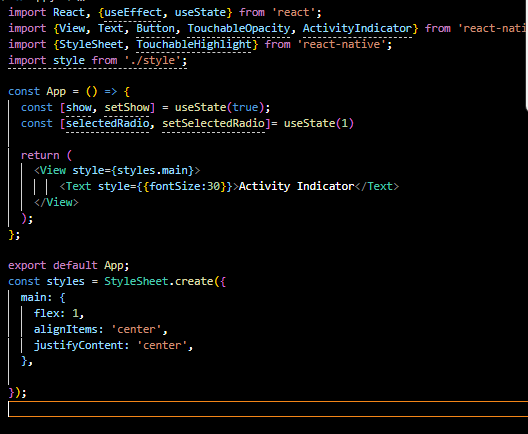
It will be like below



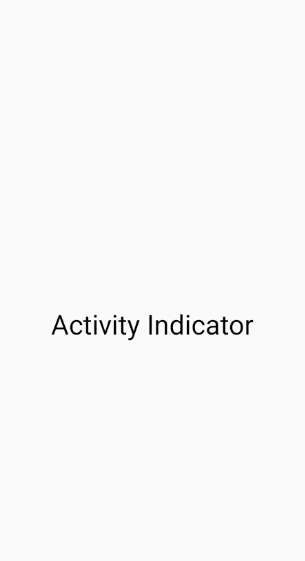
And in last some interview question will be discussed SO what we study about what is activity indicator See it is a simple loader which you used to show progression in your app. Lets say you click on a button now after clicking what happens that request for API is sent first the request for api is sent and then a response is sent from the service to the our application and then it will be displayed until then you can show the loader so that user can feel that something is happening. It is not like mobile is hanged or application is crashed such thing should not be shown so that to show an activity is being formed performed we show an activity indicator. Suppose we are fetching orders and we click on the button to start fetching it if this loader is bring shown for three seconds then it will look like that some work is going on but if this loader was not shown then you might have thought that something was supposed to happen but it is not happening and you will think that application might have been hanged or internet is working it will look like this but something is revolving or progress being shown then you will know something is going on therefore we use activity indicator. Now we talk about how to use it So first thing you need to so when adding activity indicator So import Activity indicator from react native as shown below



2:22



For now our code look like above and our output for now look like below

****

Now in our design code we replace the text component as marked below with the ActivityIndicator as shown below

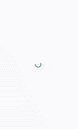


You can use it as an empty component or a container component Now you can see there is load is being shown on the mobile output as shown below

Now our design code look like below



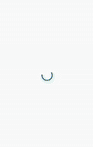
Now our loader will look like below in the center



Now you might think it is very small we need to make it slightly bigger then you can pass the prop size to the ActivityIndicator and give it value of large Now our design look like below



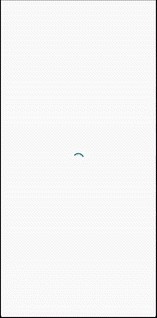
Now our mobile output look like below



You can also give medium value If you are still not satisfied with the loader size you can just enter the numerical value suppose we enter number 60 and now our design code look like below



Now our output look like below after saving the file and it will become slightly bigger as shown below

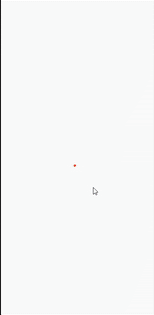


You can give any size to the loader you think you like.

So by this way you can control the size of the loader Now you might want to change its color you can do so by color props passed to the Activity indicator . We want a loader to be of red in color Now our design code look like below and we have also passed the color props there as marked below



Now our output will look like below



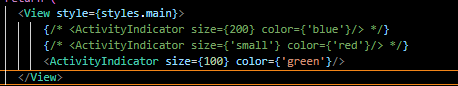
So by this you can change its color TO hide it there is a prop that animating and you ass false Boolean there then that loader will be hidden So in our design code I added animating prop and set its value to false Now our design code look like below and animating value should be passed in jsx

So now our design code looks like below

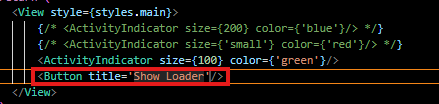


Now the screen will be blank but if you pass true then you can see the loader. Now we will make the different variants of the loader so that we can see it by different way. We make a loader of size 200 and color of blue and a loader of small size and its color is red and we comment all loaders and then we add a loader of 100 size and color of blue

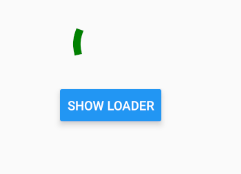
So our design code look like below



Now we want that on the button click the loader will be visible and by default it should be hidden so there are two ways to hide something Teacher will tell us both the way. So now we import a Button and we will use it inside our deign code and we give it title that is Show Loader or any name you are comfortable with Now our design code look like below



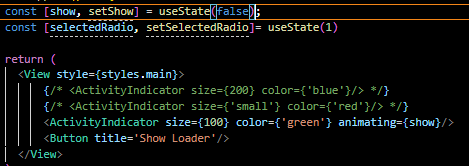
Now our output will look like below



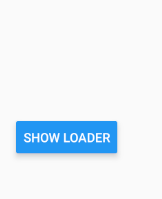
**Note:-**

I think the loader and button is a block component (who knows)

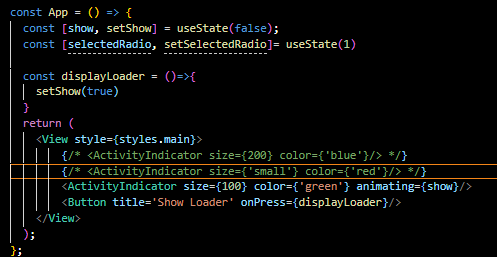
Now we make a state names show with update method like setShow and we make the default value of show as false Now we can pass the show state as the value of animating Now as we pass it then no loader is show Now our design code look like below



Now our output will look like below

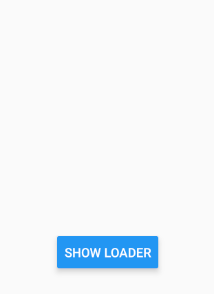


Now we make a callback function inside our App component and name it displayLoader and we will pass it to onPress of the Show loader button and then in displayLoader function we write setShow(true) to set the value of show to true Now our app component look like below

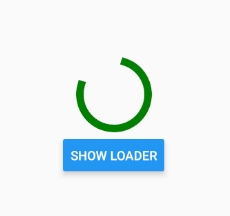


Now whenever you click on show loader button then loader is shown

Now at fist it is like below

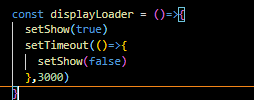


Now when you click on show loader button and then it will be visible

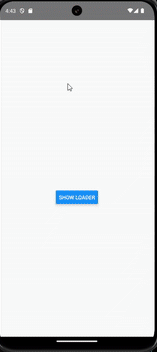


Now we want that after some time it should be removed Now suppose in displayLoader handler api is being called How we call api we will know in upcoming lesson bit from now our focus is on activity indicator Now suppose it is taking 3 second to give response to you. So in starting you will show loader and until your api does not bring result to you after it gets called till then you will only show loader until your data is loaded completely we will show loader so since we are not fetching any api so we just use setTime out for 3 second and inside it we add code setShow(false) to hide our loader

So it will be hidden after 3 seconds Now our displayLoader handler code look like below



And our output will be like below

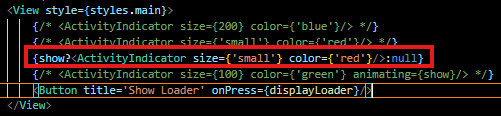


8:57

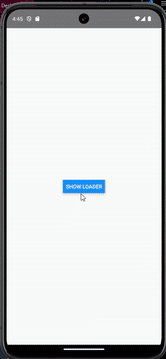
Now if we use 5 second then after 5 second our loader will disappear So now whenever we call an api then we will wait that in api how much time it is taking and as our api will reach to the client then we will hide the loader so by this way we can use loader so this was first way if you want to see other way then Teacher also showing us this that how you can hide and show the loader

The other way is similar to the way we do in previous lesson like we did by conditional operator

So we modify the design code as shown below



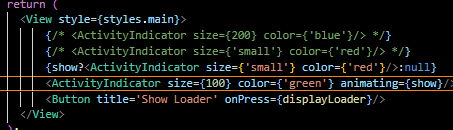
Now our output will be same as precious but this time red small slider will be shown as shown below



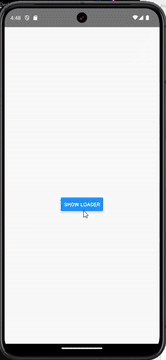
If we want we can use the first way. Teacher has told both the ways

10:05

Now we uncomment both of them and our code look like below



Now our output will look like below



Now whatever way you like

In Interview question they could ask you that can you show two loaders in the screen at once yes you can even more than two you want but usually the better UI is that which has one loader at a time we should not show more than one . Then can as you use of Activity Indicator . How to increase or decrease its size. One thing which is very important that if you want to change the size in Android then what you will do and how to bring changes in the size in IOS. Yes if you see it in an Apple phone then like we add a numeric value in size here in Android but here in Ios you need to give string value such as small , medium and large but in Android you can give numeric value and these string sizes as well So if someone ask you how to change size in IOS then there are only three sizes available that is small, medium and large and in android numerical and these three sizes are also available.