**Assignment 1**

**Group Members:**

FA20-BCS-079 Syed Muhammad Shaheer Ali Shah

FA20-BCS-090 Fazeela Rehman

URL:

<https://snack.expo.dev/@shaheershah079/assignmen1_snakesandladdersgame>

**App.js**

|  |
| --- |
| import React,{useState,useEffect} from 'react';  import { View, Text, Image, ImageBackground, StyleSheet, TouchableOpacity,  TouchableHighlight, Pressable, Button,useWindowDimensions } from 'react-native';  const App = () => {    const [dice,setDice]=useState(0);    const [leftPosition,setLeftPosition]=useState(50);    const [topPosition,setTopPosition]=useState(515);    const [leftToRight,setLeftToRight]=useState(true);    const [valI,setI]=useState(7);      const reset=()=>{      alert("You win")      setLeftPosition(50)      setTopPosition(515)      setDice(0)    }    const ladder=()=>{      if(leftPosition==120&&topPosition==445){          setTopPosition(375)          setLeftToRight(true)        }        else if(leftPosition==330&&topPosition==305){          setTopPosition(165)          setLeftPosition(260)        }        else if(leftPosition==120&&topPosition==235){          setTopPosition(95)        }    }    const snake=()=>{      if(leftPosition==260&&topPosition==375){          setTopPosition(515)          setLeftPosition(120)        }        else if(leftPosition==190&&topPosition==165){          setTopPosition(305)          setLeftPosition(260)        }        else if(leftPosition==260&&topPosition==95){          setTopPosition(305)          setLeftPosition(120)          setLeftToRight(false)        }    }    const leftMovement=()=>{      if(leftPosition!=330){        setLeftPosition(leftPosition+70)      }      else{        setTopPosition(topPosition-70);        setLeftToRight(false)      }    }    const rightMovement=()=>{      if(leftPosition!=50){        setLeftPosition(leftPosition-70)      }      else{        setTopPosition(topPosition-70);        setLeftToRight(true)      }    }    useEffect(      ()=>{        if(leftPosition==330 && topPosition==95){            reset();          }        else if(valI<=dice){          setTimeout(() => {            if (leftToRight) {              leftMovement()            }            else {              rightMovement()            }          setI(valI+1);          }, 500);        }        else{          setTimeout(() => {          ladder()          snake()          }, 500);          }      }    ,[valI]);  return (  <View style={{flex:1}}>  <ImageBackground source={require("./SnakesLaddersBoard.jpg")} style={{height: 600,}} >  <Image source={require("./player.png")}  style={[styles.playerImage, {left: leftPosition, top: topPosition}]}>  </Image>  </ImageBackground>  <View style={{flexDirection: 'row', justifyContent: 'center', alignItems:  'center'}} >  <TouchableOpacity  title="Click to throw dice"  style={styles.button}  onPress={()=>{    const rand=parseInt(((Math.random()\*5)+1))    setDice(rand)    setI(1)    }  }  >  <Text>Throw the Dice</Text>  </TouchableOpacity>  <Text> You got {dice}</Text>  </View>  </View>  );  }  const styles = StyleSheet.create(  {  button: {  padding: 10,  borderRadius: 4,  borderWidth: 1,  borderColor: 'green',  backgroundColor: 'lightgreen',  marginLeft: 5,  },  playerImage: {  width: 20,  height: 30,  }  }  )  export default App; |