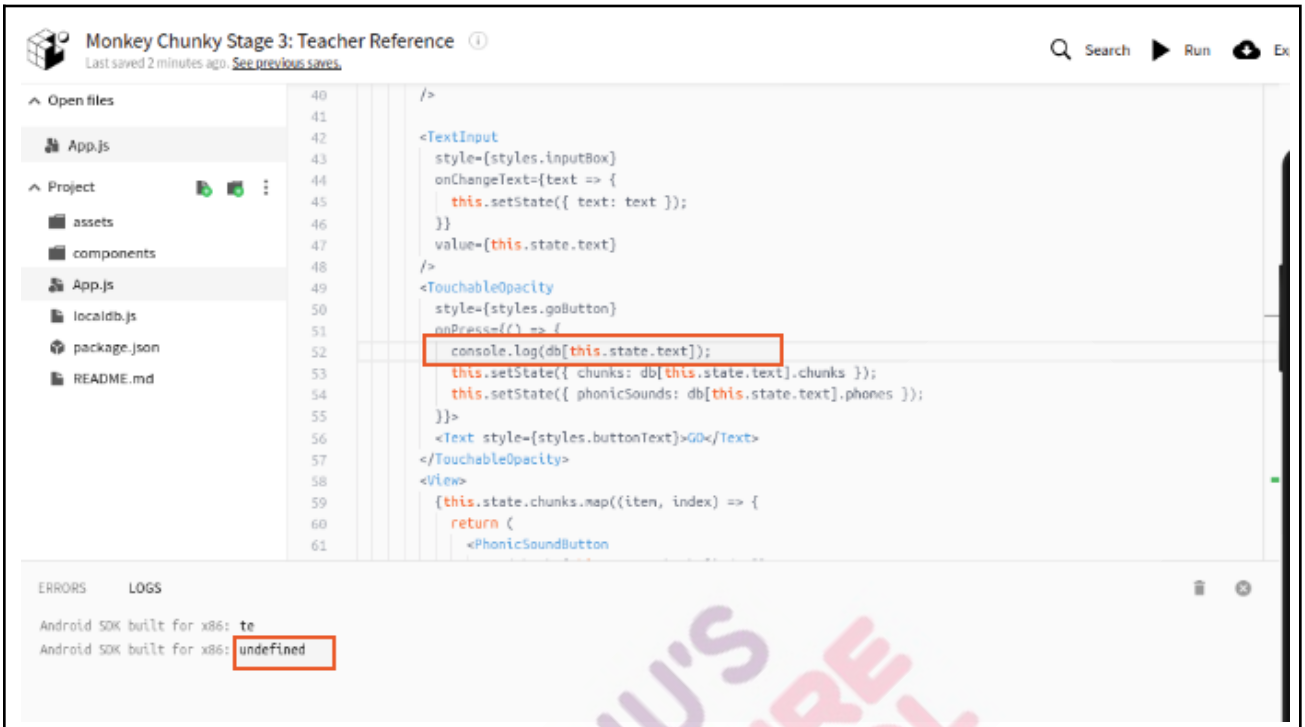
| Topic | Ternary Operators | |
|---|---|---|
| **Class Description** | **Students learn about ternary operators and how to use them in a program in place of conditional statements. They use ternary operators to add different styling for the phonic buttons when the button is pressed and displaying an alert box when there is no word in the database corresponding to the entered text by the user.** | |
| **Class** | **C66** | |
| **Class time** | **45 mins** | |
| **Goal** | • Learn about ternary operators.<br>• Use ternary operators to conditionally render different styles to the user.<br>• Fix case issue in the application. | |
| **Resources Required** | • Teacher Resources<br>  ○ Laptop with internet connectivity<br>  ○ Earphones with mic<br>  ○ Notebook and pen<br>  ○ Android/iOS Smartphone with Expo App installed<br>• Student Resources<br>  ○ Laptop with internet connectivity<br>  ○ Earphones with mic<br>  ○ Notebook and pen<br>  ○ Android/iOS Smartphone with Expo App installed | |
| **Class structure** | **Warm Up**<br>**Teacher-led Activity**<br>**Student-led Activity**<br>**Wrap up** | **5 mins**<br>**15 min**<br>**15 min**<br>**5 min** |
| **WARM-UP SESSION - 5 mins** | | |
| **Teacher starts slideshow**  **from slides 1 to 11** | | |

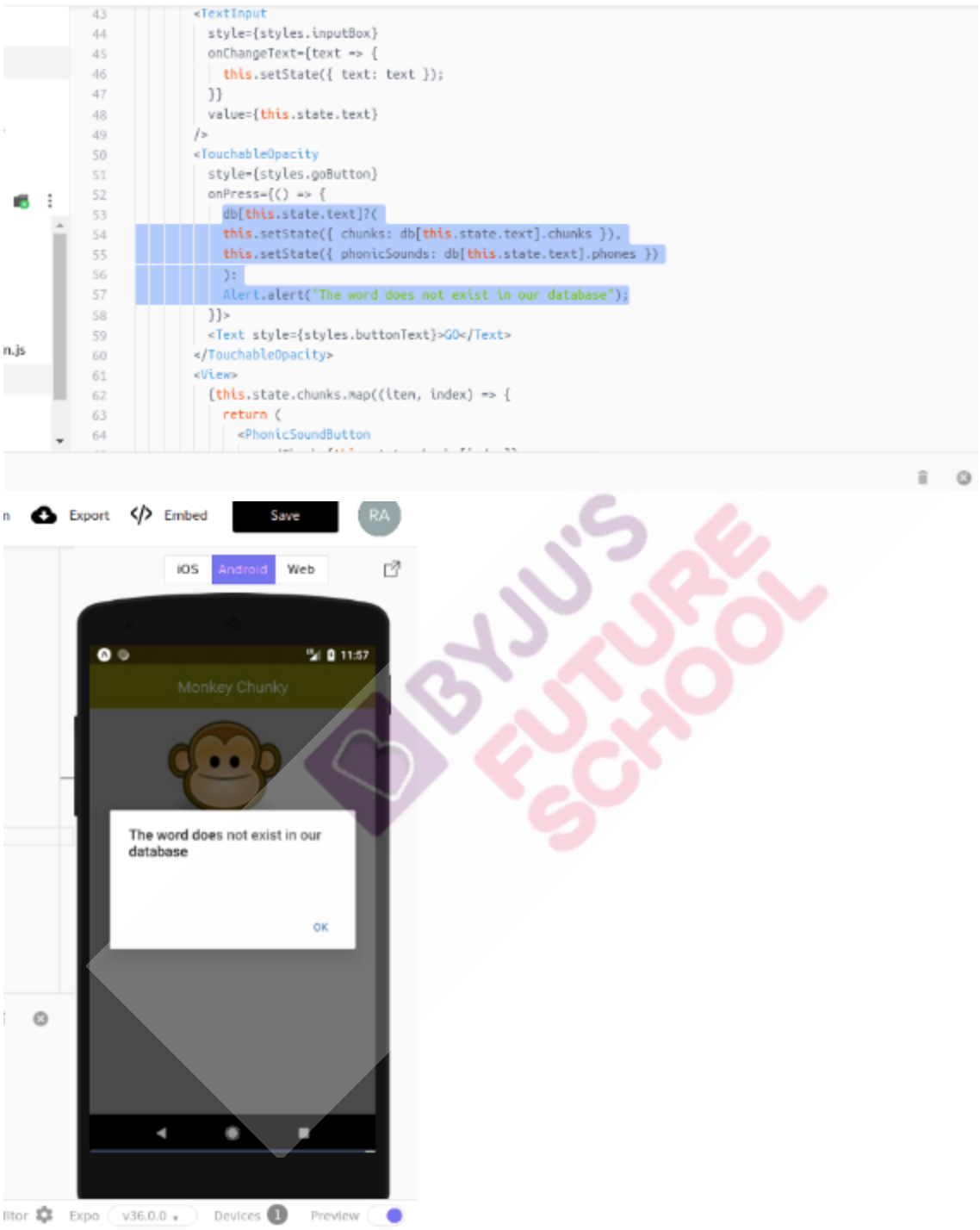
| Refer to speaker notes and follow the instructions on each slide. | |
|---|---|
| **Activity details** | **Solution/Guidelines** |
| Hi, how have you been? Are you excited to learn something new?<br><br>***Run the presentation from slide 1 to slide 4.***<br><br>**The following are the warm-up session deliverables:**<br>• Reconnect with previous class topics.<br>• Warm-Up quiz session. | **ESR**: Varied Response.<br><br>Click on the slide show tab and present the slides. |
| **QnA Session** | |
| **Question** | **Answer** |
| Which user defined component we have created in the app?<br><br>A. Touchable Opacity<br>B. Text<br>C. PhonicSound<br>D. Button | **C** |
| When are different life-cycle methods called?<br><br>A. onPress of the TouchableOpacity<br>B. when we click on the any button<br>C. the different life-cycle methods get automatically called at the different stages of the life cycle<br>D. only at the beginning of the app | **C** |
| **Continue the warm-up session** | |
| **Activity details** | **Solution/Guidelines** |

| | | |
|---|---|---|
| ***Run the presentation from slide 5 to slide 11 to set the problem statement.***<br><br>**The following are the warm-up session deliverables:**<br>• **Review code from the last class.**<br>• **Discuss issues in the Monkey-Chunky Application so far.** | | |
| **Teacher ends slideshow**  | | |
| **Teacher Initiates Screen Share** | | |
| **TEACHER-LED ACTIVITY - 15 mins** | | |
| **CHALLENGE**<br>• **Display an alert box when the word searched is not there in the database.** | | |
| **Step 2: Teacher-led Activity (15 min)** | You can quickly look at the MDN documentation of ternary operator in javascript to see how ternary operator works. | The student reads through the MDN documentation of ternary operators in **Student Activity 1**.<br>Student also looks through the examples of how ternary operators work. |
| | Can you explain how the ternary operator works? | Student explains how ternary operators work.<br>condition ? exprIfTrue : exprIfFalse |
| | Awesome!<br><br>Let's use the ternary operator to create an alert box when the typed word does not exist in the database. | ESR:<br>When the 'Go' button is pressed, we should check if |

| | | |
|---|---|---|
| | Can you guide me on how to go about this? | the word entered in the text is in the database or not. |
| | Correct.<br><br>What do we do if the word is there in the database? | ESR:<br>We store the chunks and the phones in the state using ‘this.state’. |
| | Else? | Else we display an alert Box. |
| | React Native also has an Alert component which can be used to display alert.<br><br>Let's import it.<br><br>Teacher imports Alert component from React Native library. | The student observes. |
| | | |
| | Let us quickly try to enter a word which does not exist in our database and just console what we get if we try to access it from our database.<br><br>Student writes the log message.<br><br>What do you see? | ESR:<br>‘undefined’ |




| | | |
|---|---|---|
| | We can use this as our condition for our ternary operator.<br><br>Teacher writes the code where 'db[this.state.text]' is used as a condition.<br><br>If the word exists, the state for chunks and 'phonicSounds' are set.<br>Else, an alert box is displayed.<br><br>Teacher explains how multiple statements are written separated by comma in a ternary operator. | The student observes how to use ternary operators. |




| | | |
|---|---|---|
| | Isn't this simple, short and elegant! | ESR:<br>Yes. |

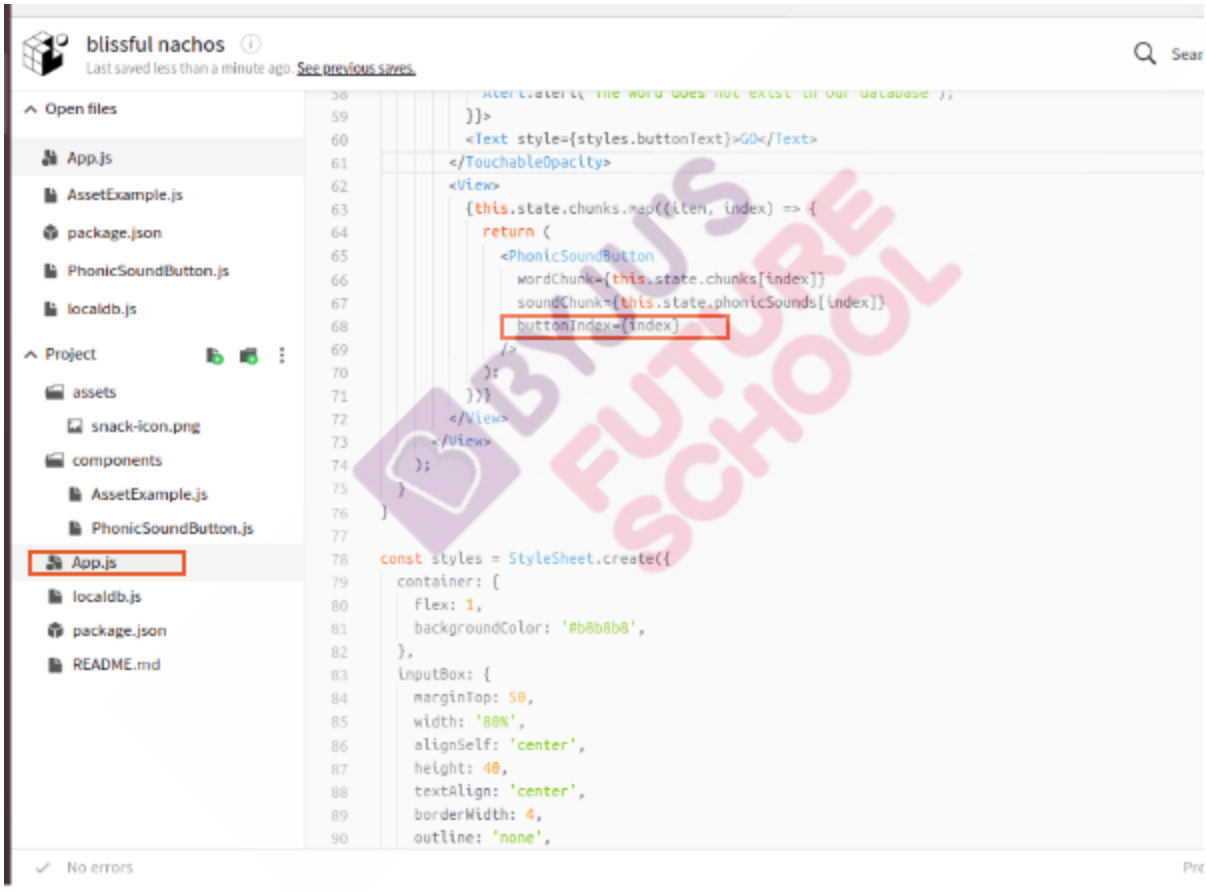
| | | |
|---|---|---|
| | Ok. Here are two challenges for you-<br><br>1. Right now our textbox search is case sensitive. If we write 'The' instead of 'the', we will get an alert that the word does not exist.<br>Can you make our text search case insensitive?<br>2. Can you use ternary operators to change the background of the phonic button when it is just pressed? This will help the user remember which button they pressed last.<br><br>Ok. Let's get started! | The student listens to the challenges and asks questions where needed. |
| **Teacher Stops Screen Share** | | |
| | Now it's your turn. Please share your screen with me. | |
| **STUDENT-LED ACTIVITY - 15 mins** | | |
| • **Ask Student to press ESC key to come back to panel**<br>• **Guide Student to start Screen Share**<br>• **Teacher gets into Fullscreen** | | |
| **ACTIVITY**<br>• **Fix case issue for text input in the app.**<br>• **Display different style for the last pressed button.** | | |
| **Teacher starts slideshow**  **for slide 12 and 13.** | | |

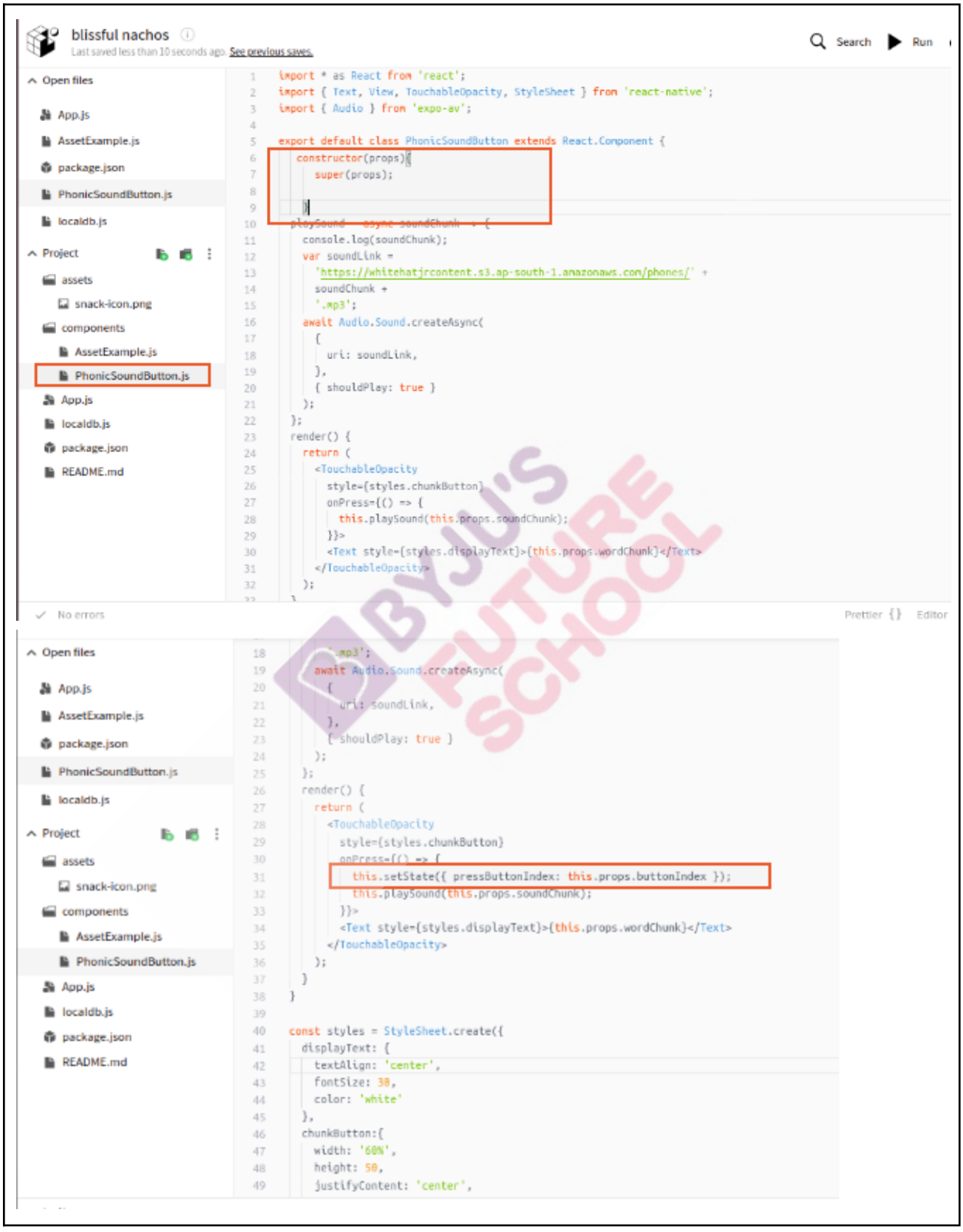
| | | |
|---|---|---|
| **Step 3: Student-Led Activity (15 min)** | Can you think about how to solve the case issue? | ESR:<br>Yes. Our database contains all the letters in lowercase. So we can convert the text from our 'TextInput' to lowercase before searching for the word in our database. |
| | Great!<br><br>Also strings already have an in-built function called '.toLowerCase()'.<br>It converts any string to lowercase.<br>You can use it to convert the text to lowercase. | The student writes code where the text is converted to lowercase first before search inside the database is performed for the word.<br><br>He/She tests the app on their phone to see if it works. |
|  | | |

| | | |
|---|---|---|
| | We can do more.<br><br>We can remove the spaces before and after the typed word by the user so that if the user has accidentally pressed spaces before or after the word, we can still find the word in the database.<br><br>String has a function called ‘trim()’ which does exactly that.<br><br>‘toLowerCase()’ returns a string so you can combine ‘.trim()’ in the same statement to strip the word of any spaces. | The student writes the code to trim the input text.<br><br>He/She tests the app on their phone to see if it works. |
|  <br> | | |
| | Now let's write code to change the style of the last phonic button which was pressed.<br><br>Any ideas on how to do that? | ESR:<br>The student tries to think about how to go about this challenge. |

| | | |
|---|---|---|
| | Allow the student to think for a few minutes. | |
| | We can create a new prop called ‘buttonIndex’ for ‘phonicSoundButton’ which passes the index of the current button.<br><br>Inside the ‘PhonicSoundButton’ component, we can have a state called ‘pressedButtonIndex’.<br><br>This will contain the index number of the word chunk button in the array which is currently pressed.<br><br>If the ‘pressedButtonIndex’ is the same as the word chunk index, we will give it one kind of style, else we will give it a different style.<br><br>Guide the student to code for this. Also explain why props need to be passed in the constructor for this component (since the component is accepting props, we need to initialize the component using these props). | The student codes as follows:<br>- He/She creates a new prop called ‘buttonIndex’ which is passed from ‘App.js’ to ‘PhonicSoundButton.js’<br>- He/She creates a new State called ‘pressedButtonIndex’ inside the ‘PhonicSoundButton.js’<br>- The student assigns the state of the ‘pressedButtonIndex’ when the button is pressed.<br>- He/She uses a ternary operator to compare if the ‘pressedButtonIndex’ state |

| | | |
|---|---|---|
| | | is the same as the ‘buttonIndex’ passed by the prop.<br>If yes, one style is used for Text and TouchableOpacity of the phonic button, else a different style (different text and background color) is used. |






| | | |
|---|---|---|
| | Great work! | |

**Teacher Guides Student to Stop Screen Share**

**WRAP-UP SESSION - 5 Mins**

**Teacher starts slideshow from slide 14 to slide 23**

| Activity details | Solution/Guidelines |
|---|---|
| **Run the presentation from slide 14 to slide 23**<br><br>**Following are the wrap-up session deliverables:**<br>• **Explain the facts and trivias**<br>• **Next class challenge**<br>• **Project for the day**<br>• **Additional Activity** | Guide the student to develop the project and share with us. |

**Quiz time - Click on in-class quiz**

| Question | Answer |
|---|---|

<table>
<tr><td>Which of the following is the correct syntax for the ternary operator?<br>A. condition ? exprIfFalse : exprIfTrue<br>B. condition ? exprIfTrue : exprIfFalse<br>C. condition : exprIfTrue ? exprIfFalse<br>D. condition ? exprIfTrue ; exprIfFalse</td><td colspan="2">B</td></tr>
<tr><td>Which of the following statements is used to display an alert?<br>A. alert(“message”)<br>B. Alert(“message”)<br>C. alert.message<br>D. alert.message(“text”)</td><td colspan="2">A</td></tr>
<tr><td>Which of the following functions is used to convert a string to lowercase?<br>A. toLower()<br>B. lowerCase()<br>C. ToLowerCase()<br>D. toLowerCase()</td><td colspan="2">D</td></tr>
<tr><td colspan="3"><u>FEEDBACK</u><br>• Encourage the student to read about ternary operators.</td></tr>
<tr><td></td><td>You get a “hats off”.<br><br>Till next class then. See you. Bye!</td><td>Make sure you have given at least 2 Hats Off during the class for:<br>  <br>  <br>  </td></tr>
<tr><td>Project Pointers and Cues (5 min)</td><td>TEXT TO SPEECH - TERNARY OPERATOR</td><td></td></tr>
</table>

**Goal of the Project:**

Today you learned about ternary operators and how to use them in a program in place of conditional statements.

In this project you need to use a ternary operator in text to speech converter app on which you worked in project 65. This will be an additional feature in text to speech converter apps.

*This is a continuation of Project 65. So make sure to complete that project before you attempt this one.*

**Story:**

Saisha's friend Lisa is visiting her from France. She understands English, but speaks only French.

I am very excited to see your project solution and I know you both will do really well.

Bye Bye!

**Teacher ends slideshow** 

**Teacher Clicks** ✖ End Class

| | | |
|---|---|---|
| **Additional Activities** | Encourage the student to look at additional documentation for ternary operators and various string methods. | |
| | Encourage the student to write reflection notes in their reflection journal using markdown.<br><br>Use these as guiding questions:<br><br>● What happened today?<br>- Describe what happened<br>- Code I wrote<br>● How did I feel after the class?<br>● What have I learned about programming and developing games?<br>● What aspects of the class helped me? What did I find difficult? | The student uses the markdown editor to write her/his reflection in a reflection journal. |

| Activity | Activity Name | Links |
|---|---|---|
| Teacher Reference | Final Reference | https://snack.expo.io/@rajeevtfi/efeeee |
| Student Activity 1 | Ternary MDN documentation | https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Conditional_Operator |

| | | |
|---|---|---|
| Student Activity 2 | Class Activity | https://snack.expo.io/@rajeevtfi/monkey-chunky-stage-3:-teacher-reference |
| Student Activity 3 | Ternary Operators more info | https://scotch.io/tutorials/understand-the-javascript-ternary-operator-like-abc |
| Student Activity 4 | String methods | https://www.w3schools.com/js/js_string_methods.asp |
| Teacher Reference visual aid link | Visual aid link | https://curriculum.whitehatjr.com/Visual+Project+Asset/PRO_VD/BJFC_PRO_V3_C66_withcues.html |
| Teacher Reference In-class quiz | In-class quiz | https://s3-whjr-curriculum-uploads.whjr.online/3a1320de-170e-4a41-b56c-5cc05c33a554.pdf |