

Topic	VIEW STATES				
Class Description	Students learn to add the A-Frame cursor click event. Students will learn to change state variables' value on cursor click events to show and hide elements on click events.				
Class	C159				
Class time	45 mins				
Goal	<ul style="list-style-type: none"> ● Add different state variables to the scene. ● Show and hide elements based on different states. ● Add a cursor click event to handle different states. 				
Resources Required	<ul style="list-style-type: none"> ● Teacher Resources <ul style="list-style-type: none"> ○ Visual Studio Code Editor ○ laptop with internet connectivity ○ earphones with mic ○ notebook and pen ● Student Resources <ul style="list-style-type: none"> ○ Visual Studio Code Editor ○ laptop with internet connectivity ○ earphones with mic ○ notebook and pen 				
Class structure	Warm-Up Teacher-led Activity Student-led Activity Wrap-Up		05 mins 15 mins 20 mins 05 mins		
WARM-UP SESSION - 05 mins					
<u>CONTEXT</u>					
<ul style="list-style-type: none"> ● Add different state variables to show and hide elements in the scene. 					


Teacher starts slideshow from slides 1 to 15

Refer to speaker notes and follow the instructions on each slide.

Activity details	Solution/Guidelines
<p>Hi, how have you been? Are you excited to learn something new?</p> <p>Run the presentation from slide 1 to slide 3.</p> <p>The following are the warm-up session deliverables:</p> <ul style="list-style-type: none"> • Reconnect with previous class topics. • Warm-Up quiz session. 	<p>ESR: Varied Response.</p> <p>Click on the slide show tab and present the slides.</p>
Q&A Session	
Question	Answer
<p>How can we change the color of the selected thumbnail ring?</p> <p>A. using setAttribute() B. using setThumbnail() C. using setThumbnailring() D. using setAttributeNode()</p>	<p>A</p>
<p>Which of the following did we use to check for the selected thumbnail's id?</p> <p>A. handleList() B. handleThumbnail() C. handlePlaceListState() D. handleID()</p>	<p>C</p>
Activity details	Solution/Guidelines

<p>Run the presentation from slide 4 to slide 15 to set the problem statement.</p> <p>The following are the warm-up session deliverables:</p> <ul style="list-style-type: none"> • A-Frame Physics System • Static and dynamic bodies 		
<div style="background-color: #9ACD32; color: white; padding: 10px; text-align: center;">  Teacher ends slideshow </div>		
<div style="background-color: #008000; color: white; padding: 10px; text-align: center;"> Teacher Initiates Screen Share </div>		
	TEACHER-LED ACTIVITY - 15 mins	
<div style="background-color: #FADBD8; color: black; padding: 10px; text-align: center;"> <u>CHALLENGE</u> <ul style="list-style-type: none"> • Add State to the different actions of the scene. • Write functions to change the state on click. </div>		
Step 2: Teacher-led Activity (15 mins)	<i><The teacher opens the code from the previous class.></i>	
	<p>We want to start the tour when the user clicks on one of the thumbnails.</p> <p>And how can we do that as we have so much content on our page?</p>	ESR: Varied!
	<p>To do so we can hide all the elements from the screen and just have the image of the selected thumbnail on the screen when the user clicks on the screen.</p>	
	<p>Let's divide what we want to do in different states:</p>	

	<ul style="list-style-type: none"> The places-list state which would be the initial screen and have all the elements. The “view” state where we'll be seeing the image of the clicked thumbnail. 	
	<p>Let's start by creating these states first.</p> <p>As the places-list state is going to be initial, let's store it in the schema.</p> <p><i><In the Tour.js file, the teacher codes to create the schema and creates a variable called states and default value as places-list.></i></p> <p>Now we'll write the function to hide the elements on the screen.</p> <p>Can you tell me how we can hide the elements on the screen?</p> <p>Awesome! We'll call this function as hideEl. In this function, using the map function, we'll get all the elements from the list and set its visible attribute to false.</p> <p>Note: Teacher can help</p> <p><i><The teacher codes to create a hideEl</i></p>	<p>ESR:</p> <p>We can set the visible attribute to false for the entities to make them disappear.</p>

	<p><i>function. Inside the function, using the map function, get the list of the elements and set their visible attribute to false.></i></p> <pre>Tour.js > ⚡ schema AFRAME.registerComponent("tour", { schema: [state: { type: "string", default: "places-list" },], hideEl: function(elList) { elList.map(el => { el.setAttribute("visible", false); }); }, },</pre>	
	<p>Next we want to show the tour image of the selected thumbnail.</p> <p>Normally when we visit the place, what view do we have of that place?</p> <p>What can we do to give the same feeling on our tour?</p> <p>Awesome, we have used cards to show the thumbnails and now we will build the 360° image gallery of the selected card.</p> <p>For this we need to have 360° images, which are wide-angle view images spanning the full circle. These can be downloaded from different websites.</p>	<p>ESR: No image.</p> <p>ESR: We have a 360-degree view of the place.</p> <p>ESR: We can add the 360-degree images to the tour.</p>

	<p style="color: red;"><i><The teacher codes to create a selectedCard variable in the schema and sets #card1 as the default value.></i></p> <pre>AFRAME.registerComponent("tour", [schema: { state: { type: "string", default: "places-list" }, selectedCard: { type: "string", default: "#card1" }, },],</pre>	
	<p>Now, let's write a function called showView() which will show the image of the selected card.</p> <p>Can you tell me how can we do that?</p> <p>Perfect!</p> <p>To do so :-</p> <ul style="list-style-type: none"> • Get the data from the schema and store it in the selectedCard variable using the document.querySelector(). • Select the main container and set it in the skyEl variable. • Set the image of the selected card, using the material attribute. <p style="color: red;"><i><The teacher codes to create a function called as showView function.></i></p>	<p>ESR:</p> <p>We can get the image of the selected card and set it on the sky entity.</p>

```

showView: function() {
    const { selectedCard } = this.data;
    const skyEl = document.querySelector("#main-container");
    skyEl.setAttribute("material", {
        src: `./assets/360_images/${selectedCard}/place-0.jpg`,
        color: "#fff"
    });
},
    
```

	<p>How can we know if the state is updated or not?</p> <p>We need to keep checking for the change in the state continuously. A-Frame provides us with a function called .tick() function; this function calls itself continuously.</p> <p>Inside this function, we can check if the state is changed to view. If the state is changed, we will call the <code>hideEl()</code> function, pass the places container list, and call the <code>showView()</code> function.</p> <p><i><The teacher codes to create the tick function.></i></p>	ESR: Varied!
--	--	--------------

```

tick: function() {
    const { state } = this.el.getAttribute("tour");

    if (state === "view") {
        this.hideEl([this.placesContainer]);
        this.showView();
    }
},
    
```

	<p>We have everything ready on what to hide and what to show. But when do we want all this to happen?</p>	<p>ESR: This should happen when we click on the thumbnails.</p>
	<p>Yes, and what are the things that we want to do when we click the thumbnail?</p> <p>Awesome. Let's write a function which will do this for us.</p> <p>We'll call the function as handleClickEvents().</p> <p>Can you try to add this functionality in your code?</p> <p>I'll guide you where you get stuck.</p>	<p>ESR: We want to check which thumbnail is clicked and based on that, set the image to the sky entity.</p> <p>ESR: Yes!</p>
Teacher Stops Screen Share		
	<p>Now it's your turn. Please share your screen with me.</p>	
STUDENT-LED ACTIVITY - 15 mins		
<ul style="list-style-type: none"> Ask the student to press the ESC key to come back to the panel. Guide the student to start screen share. Teacher gets into fullscreen. 		
<u>ACTIVITY</u>		
<ul style="list-style-type: none"> Write the code by revising the concept of states. 		
<p> Teacher starts slideshow from slides 16 and 17</p>		

Refer to speaker notes and follow the instructions on each slide.		
Step 3: Student-Led Activity (15 mins)	<p><Teacher guides the student to clone/download boilerplate code from the [Student Activity-1].></p>	<p><Student opens code from the previous class.> [Student Activity-1]</p>
	<p>Now, we'll write the handleClickEvents() function, which will handle the cursor click events for us.</p> <p><Teacher guides student to code for the handleClickEvents() in CursorEvent.js File.></p>	<p><Student codes to write the handleClickEvents() function.></p>
<p>Note: Below code is part of the Student Boilerplate Code</p> <pre> JS CursorEvents.js > ⚡ handleClickEvents AFRAME.registerComponent("cursor-listener", { schema: { selectedItemId: { default: "", type: "string" } }, init: function () { this.handleMouseEnterEvents(); this.handleMouseLeaveEvents(); }, handleClickEvents: function () { //Cursor 'click' Events /* */ ADD CODE HERE /* } },</pre>		

	<p>In this function, first we need to check for clicks. So we'll add an event listener which will listen to all the mouse click events.</p> <p>Inside the listener function, using the <code>document.querySelector()</code> method, we'll get the places-container and store it in the places container variable.</p> <p>Using the get attribute, we'll get the state from the tour entity.</p> <p>If the state is places-list then using the get attribute we'll get the id. Create a placedId list with all places id in it.</p> <p>Then we check if the placesId list contains the id that we have, if it does, then we set the state to view and set the id to <code>selectedCard</code> variable.</p> <p style="color: red;"><i><The teacher guides student to write the code for the <code>handleClickEvents()</code> function and call it in <code>init()</code> function.></i></p>	
--	---	--

```

handleClickEvents: function () {
    //Cursor 'click' Events
    this.el.addEventListener("click", evt => {
        const placesContainer = document.querySelector("#places-container");
        const { state } = placesContainer.getAttribute("tour");

        if (state === "places-list") {
            const id = this.el.getAttribute("id");
            const placesId = [
                "taj-mahal",
                "budapest",
                "new-york-city",
                "eiffel-tower"
            ];
            if (placesId.includes(id)) {
                placesContainer.setAttribute("tour", {
                    state: "view",
                    selectedCard: id
                });
            }
        }
    });
}
    
```

```

AFRAME.registerComponent("cursor-listener", {
    schema: {
        selectedItemId: { default: "", type: "string" }
    },
    init: function () {
        this.handleClickEvents();
        this.handleMouseEnterEvents();
        this.handleMouseLeaveEvents();
    },
}
    
```

	Let's run the code to check the output.	<i><Student runs the code to check the output.></i>
--	---	---

It's time to travel virtually!

Travel Around The World



Taj Mahal



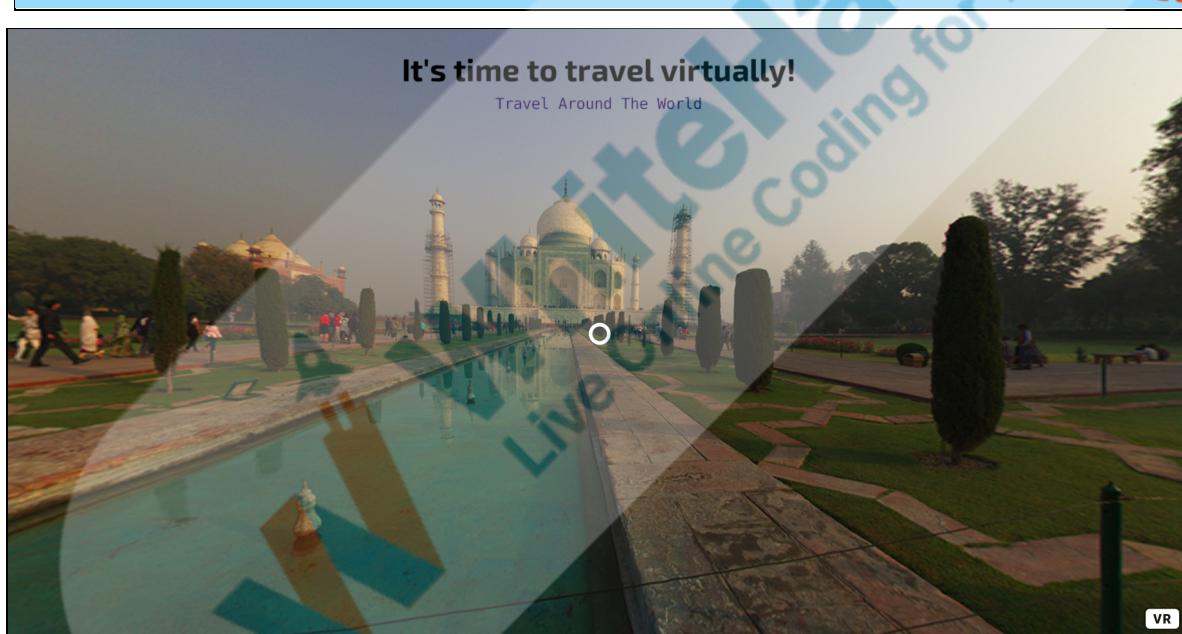
Budapest



Eiffel Tower

It's time to travel virtually!

Travel Around The World



Teacher Guides Student to Stop Screen Share

WRAP-UP SESSION - 5 Mins

FEEDBACK

- Compliment the student for her/his effort in the class.
- Encourage the student to think and come up with their own solutions.

 Teacher starts slideshow from slide 18 to slide 29	
Activity details	Solution/Guidelines
Run the presentation from slide 18 to slide 29 Following are the wrap-up session deliverables: <ul style="list-style-type: none"> • Explain the facts and trivias • Next class challenge • Project for the day • Additional Activity 	Guide the student to develop the project and share with us.
Quiz time - Click on in-class quiz	
Question	Answer
How can we hide the elements on the screen? A. by setting the visibility attribute of the function to false B. by setting the visibility attribute of the function to true C. by setting the visibility attribute of the function to null D. function hideEl	D
360° images are: A. narrow-angle view images spanning the full circle B. wide-angle view images spanning the semi-circle C. wide-angle view images spanning the full circle D. narrow-angle view images spanning the semi-circle	C
Which of these is used to show the image of the selected card? A. showviews() B. showView() C. showCard() D. showScreen()	B
• End the quiz panel	

	<p>You get a “hats-off”.</p> <p>Alright. See you in the next class.</p>	<p><i>Make sure you have given at least 2 Hats Off during the class for:</i></p> <div style="display: flex; justify-content: space-around; margin-top: 10px;"> <div style="text-align: center;">  <p>Creatively Solved Activities +10</p> </div> <div style="text-align: center;">  <p>Great Question +10</p> </div> <div style="text-align: center;">  <p>Strong Concentration +10</p> </div> </div>
Project Overview	<p>COMIC WORLD STAGE 3</p> <p>Goal of the Project:</p> <p>In this project you will add a cursor click event to change the thumbnail cards view state to show the story text of the selected card for the project created in the previous class.</p> <p>Story:</p> <p>Your friend always wanted to watch DC & MARVEL movies, but he is not getting time to watch them. He always wished to go through their comic stories online whenever he got time.</p> <p>Help him to create a virtual comics world and add the particular comics for thumbnail components for the tour to comics. Add more elements to the scene and help your friend</p>	

	<p>see different thumbnails' shapes, and add your favorite comic's posters in the thumbnails.</p> <p>Write an A-Frame component to add the cursor click event in the scene to select the entity on mouse click and show a story of the selected comic book.</p> <p>I am very excited to see how you would add a cursor click event in the scene.</p> <p>Bye!</p>	
Additional Activities	<p><i>Encourage the student to add the zoom in and zoom out functionality when the arrow key is pressed.</i></p> <p>Update the “tour” component in the Tour.js file.</p> <ul style="list-style-type: none"> • Add the “zoomAspectRatio” variable in the schema. 	<p><i>The student observes and learns.</i></p>
<pre>AFRAME.registerComponent("tour", { schema: { state: { type: "string", default: "places-list" }, selectedCard: { type: "string", default: "#card1" }, zoomAspectRatio: { type: "number", default: 1 } }, })</pre>		

	<ul style="list-style-type: none">• Add the keydown event handler in <code>.update()</code> life handler method of the component.• Check if the Up Arrow or Down Arrow key is pressed.• Add the condition to check the state and zoom ratio.• Increase the value of the zoom factor when the Up Arrow key is pressed and set the camera element zoom attribute.• Decrease the value when the Down Arrow key is pressed and the camera element zoom attribute.	
--	---	--

```

update: function() {
    window.addEventListener("keydown", e => {
        if (e.key === "ArrowUp") {
            if (
                (this.data.zoomAspectRatio <= 10 && this.data.state === "view") ||
                (this.data.zoomAspectRatio <= 10 && this.data.state === "change-view")
            ) {
                this.data.zoomAspectRatio += 0.002;
                this.cameraEl.setAttribute("zoom", this.data.zoomAspectRatio);
            }
        }
        if (e.key === "ArrowDown") {
            if (
                (this.data.zoomAspectRatio > 1 && this.data.state === "view") ||
                (this.data.zoomAspectRatio > 1 && this.data.state === "change-view")
            ) {
                this.data.zoomAspectRatio -= 0.002;
                this.cameraEl.setAttribute("zoom", this.data.zoomAspectRatio);
            }
        }
    });
},
    
```

Activity	Activity Name	Links
Teacher Activity 1	Previous Class Code	https://github.com/whitehatjr/PRO-C_158-Teacher-Ref
Teacher Activity 2	Teacher Reference Code	https://github.com/whitehatjr/PRO-C_159-Teacher-Ref
Student Activity 1	Student Boilerplate	https://github.com/whitehatjr/PRO-C_159-Student-Boilerplate
Student Activity 2	360° Image Assets	https://github.com/whitehatjr/PRO-C_159-Student-Boilerplate/tree/main/assets/360_images
Project Solution	Comic World Stage 3	https://github.com/whitehatjr/PRO-C_159-Project-Solution
Teacher Ref. Visual	Visual Aid link	https://curriculum.whitehatjr.com/Vis

Aid Link		<u>ual+Project+Asset/PRO_VD/PRO_C159_withcues.html</u>
Teacher Ref. In-Class Quiz	In-Class Quiz	<u>https://s3-whjr-curriculum-uploads.whjr.online/3dc5d195-57d3-4fb7-becf-072873c248b4.pdf</u>