

KEY IN-CLASS STUDENT ACTIVITY 3.2B: PLAN FOR GAME STATE

Lesson Objective 3.2:

Plan for game state. *Topics:* understanding games' main loop (input/update/render), graphics pipeline; understanding the flow of a game, loading, menus, save-load.

Resources, software, and additional files needed for this lesson:

1. A workstation with Microsoft Office 2007® or 2010 installed
2. Microsoft Visual Studio 2010® and XNA® 4.0
3. Internet access

Guiding questions:

1. **What is a scene hierarchy engine and how does it work?** Scene hierarchy engine is another term for the code that is responsible for game state management. The ScreenManager class is a reusable component that maintains a stack of one or more GameScreen instances. It coordinates the transitions from one screen to another, and takes care of routing user input to whichever screen is on top of the stack.
2. **What does save-load mean?** Save-load is used for maintaining the game state for the user. The data is saved and loaded back in when the user resumes game play.
3. **How do you determine the gamepad state?** GamePadState currentState = GamePad.GetState(PlayerIndex.One); This call to GetState retrieves a GamePadState object, which contains the information we need about the controller.

Student Activity:

Directions to the student:

1. Download the project file and extract it to its own directory. Download the CPModel_Sample.zip file from <http://msdn.microsoft.com/en-us/library/bb203933.aspx>.
2. Complete the tutorial: "Rendering a Model with a Basic Effect." Incorporate the specifications indicated in the scenario. Show your instructor the completed projects when finished. Answer the questions.

Scenario:

You've learned a great deal about designing games in the Game Design Principles class at the community college. In the next assignment, you have been assigned to create a spaceship game. The games must accept user input to move the ship across the screen and meet these specifications for moving the model:

1. Pressing left or right on the left thumbstick will move the model in the X or Y direction.
2. Pressing the DPad up or down will change the X and Y rotation.
3. Pressing the A button will exit the program.

Download CPModel_Sample.zip

<http://msdn.microsoft.com/en-us/library/bb203933.aspx>

Content:

1. Summarize the steps you completed to create the project in the tutorial.

Answers will vary.

2. Did you have any problems creating the project? If so, how did you solve them?

Answers will vary.

3. What did you learn by completing the tutorial and creating the project?

Answers will vary.

4. How does what you learned relate to the objective of this lesson, "Plan for game state"?

Answers will vary.