Grading Rubric – NP HTML and CSS, 8e
Tutorial 8, Case Problem 2

Class:
Professor:
Notes:

Description	Pts	Your Score
1. Using your editor, open the paa_game_txt.html and paa_animate_txt.css files from the html08 > case2 folder. Enter your name and the date in the comment section of each file, and save them as paa_game.html and paa_animate.css respectively.	**1**	
2. Go to the paa_game.html file in your editor. Add a link to the paa_animate.css style sheet file to the document head.	**2**	
3. Scroll down to the gameBox div element. Within this element, insert three div elements with the ids butterfly, bat, and fox and belonging to the sprite class. These div elements will contain animated backgrounds showing three characters from the game.	**3**	
5. Within the Transition Effects section, insert a style rule for the nav#gameLinks a selector that: a) places the links using relative positioning, b) sets the font color to white, and c) transitions the font color over a 0.5-second interval.	**6**	
6. Insert a style rule for the nav#gameLinks a:hover selector that sets the font color to rgb(255, 194, 99).	**2**	
7. Heather wants a transition effect applied to the links in the gameLinks list in which a gradient-colored bar gradually expands under each link during the hover event. To create this effect, you will use the after pseudo-element and the content property to insert the bar. Create a style rule for the nav#gameLinks a::after selector that: a) places an empty text string as the value of the content property, b) places the content with absolute positioning with a top value of 100% and a left value of 0 pixels, c) sets the width to 0% and the height to 8 pixels, d) changes the background to a linear gradient that moves to right from the color value rgb(237, 243, 71) to rgb(188, 74, 0), e) sets the border radius to 4 pixels, and f) hides the bar by setting the opacity to 0.	**12**	
8. When the links are hovered over, change the appearance of the bar by adding a style rule for the nav#gameLinks a:hover::after selector that changes the opacity to 1 and the width to 100%.	**2**	
9. Return to the style rule for the nav#gameLinks a::after selector and add a transition style that applies the opacity and width changes over a half-second interval.	**2**	

11. Return to the paa_animate.css file and, within the Sprite Styles section, create a style rule that displays all div elements of the sprite class with absolute positioning.	**2**	
12. For the div element with the ID bat, create a style rule that: a) sets the width and height to 40 pixels by 50 pixels, b) sets the top and left coordinates to 100 pixels and -50 pixels, and c) displays the paa_bat.png as the background image placed at the left center of the background with no tiling and sized to cover the background.	**6**	
13. Create a similar style rule for the div element with the ID butterfly, setting the width and height at 35 pixels, the top-left coordinates at 60 pixels and -50 pixels, and using the paa_bfly.png as the background image. Create another style rule for the div element with the ID fox, setting the width and height at 280 and 260 pixels, the bottom and right coordinates at 10 pixels, and the paa_fox.png file as the background image. (Note: The background image in all aminations should place the image at the left center with no tiling and sized to cover the background.)	**4**	
14. Sprites are animated by moving the background image file across the background of the object. Go to the Animation Styles section and create an animation named playSprite that sets the background image position to right center at 100% time.	**2**	
15. Heather wants the bat and butterfly to flutter as they move across the animation box. Create an animation named flyRight with the following key frames: a) at 25% time, set the top coordinate to 150 pixels, b) at 50% time, set the top coordinate to 55 pixels, c) at 65% time, set the top coordinate to 120 pixels, d) at 90% time, set the top coordinate to 50 pixels, and e) at 100% time, set the top and left coordinates to 80 pixels and 100%.	**6**	
16. Sprites achieve the animation effect by changing the background image in n - 1 discrete steps, where n is the number of frames in the sprite. Apply the playSprite animation to the fox div element after a 4-second delay over a time interval of 3.5 seconds and a steps value of 27. Set the animation to loop infinitely.	**6**	
17. Apply the playSprite animation to the bat div element over a 2-second interval with 39 steps. Apply the flyRight animation over an 8-second interval with linear timing. Set both animations to loop infinitely.	**4**	
18. Apply the playSprite animation to the butterfly div element after a 3-second delay, with a playing time of 1 second spaced out in 33 steps. Apply the flyRight animation over a 6-second interval. Make the butterfly appear to hover by applying a Cubic Bezier curve to the flyRight timing with the function cubic-bezier(0,1,0.73,0). Set both animations to loop infinitely.	**4**	
20. Hover your mouse pointer over the four links below the Frustrated Fox logo and verify that a gradient-filled bar grows beneath the links in response to the hover event.	**1**	
21. Verify that the animation box shows an animated bat and then a butterfly moving across the sky and that, after a short delay, an animated fox jumps up toward the bat and butterfly trying to catch them.	**1**	
TOTAL	**66**	

YOUR SCORE: **______**