Übung JavaScript Shape Canvas

Ausgangslage

Diese Übung basiert auf der Übung Rectangle Canvas Method. Zusätzlich sollen 2 Ovale gezeichnet werden:Zuerst kopieren wir die Oval Klasse aus dem bestehenden Rectangle und passen die getArea() und draw(...) Methode an, so dass ein Oval berechnet und gezeichnet wird.Das folgende Script zeigt auf wie man Ovale mit dem HTML Canvas zeichnen kann:// fill oval
ctx.ellipse(this.x, this.y, this.width / 2, this.height / 2, 0, 0, Math.PI*2)
ctx.fill();// stroke oval
ctx.ellipse(this.x, this.y, this.width / 2, this.height / 2, 0, 0, Math.PI*2)
ctx.stroke();Die Klassen Rectangle und Shape arbeiten mit den gleichen Attributen. Solche können wir über die Superklasse Shape auslagern.Das folgende UML Diagramm zeigt dies auf:

Vorgehen

Lösen Sie bitte die Aufgabe wie folgt:Kopieren Sie die Datei Rectangle.js um nach Oval.js. Benennen Sie in der Datei die Klasse nach Oval um.Passen Sie die Methode Oval.getArea() um so dass die Fläche eines Ovals berechnet wird (h*b * PI / 4).Passen Sie die bestehende index.html Lösung so an, dass neu 2 Ovale gezeichnet werden.Das folgende Listing zeigt das HTML JavaScript:let myCanvas = document.getElementById('myCanvas');let o1 = new Oval(220, 70, 100, 50, '#FFFFFF');
o1.draw(myCanvas);
let o2 = new Oval(200, 300, 50, 100, '#FF00FF', true);
o2.draw(myCanvas);Testen Sie das Resultat via Run -> Chrome Browser.Erstellen Sie die Datei Shape.js und definieren Sie die Klasse Shape mit den gleichen Attributen gemäss der Klasse Rectangle oder Oval.Entfernen Sie in der Rectangle und Oval Klasse die Attribute und leiten Sie aus der Klasse Shape ab. Programmieren Sie den constructor um.Testen Sie das Resultat via Run -> Chrome Browser, es sollte von der Anzeige immer noch identisch sein.

Lösung

Eine mögliche Lösung finden Sie hier

Kontakt

Simtech AG
Finkenweg 23
3110 Münsingen
Schweiz

Impressum

Das Copyright für sämtliche Inhalte dieser Website liegt bei Simtech AG, Schweiz.
Beachten Sie auch unsere Hinweise zum Urheberrecht, Datenschutz und Haftungsauschluss.
Jeder Hinweis auf Fehler nehmen wir gerne entgegen.

Copyright

2024 Simtech AG, All rights reserved, Powered by stack.ch written in Golang by Daniel Schmutz