LU01.L02 - myGeometryCalculator

Auftrag

Programmieren Sie einen Rechner *myGeometryCalculator.js*, der die genannten geometrischen Funktionen ausführen kann. Testen Sie anschliessend Ihre Lösung auf Funktion, indem Sie diese ausführen und das Ergebnis auf der Kommandozeile ausgeben lassen.

Lösung 1: Kreisfläche

```
const pi = 3.14; //accuracy is sufficient enough for our purpose
let r = 10; // 10 is easy to calculate

function circleArea(radius) { // Variante ohne lokale Veriable
    return pi*radius*radius;
}

function circleArea2(radius) {
    let cA = pi*radius*radius; // mit lokaler Variable
    return cA;
}

console.clear(); // Terminal wird bereinigt
console.log("function circleArea mit Radius: ", r, " = ", circleArea(r),
"Flaecheneinheiten.");
console.log("function circleArea2 mit Radius: ", r, " = ", circleArea2(r),
"Flaecheneinheiten.");
```

Teilauftrag 2: Kreisumfang

```
function circleCircumference(radius) { // Variante ohne lokale Variable
    return 2*pi*radius;
}

function circleCircumference2(radius) { // Variante mit lokaler Variable
    let area = 2*pi*radius;
    return area;
}

console.log();
console.log("function circleCircumference: " + r + " = " +
circleCircumference(r));
console.log("function circleCircumference2: " + r + " = " +
circleCircumference2(r));
```

Teilauftrag 3: Dreiecksfläche

```
function triangleArea(side, height) {
    let area = side*height/2; // Lokale Variable ist nur sichtbar innerhalb der Funktion
    return area;
}

console.log();
/* Aus Gründen der Übersichtlichkeit wird nachfolgend nur eine Variante verwendet: Zwischenschritt mit
   mit Speicherung in lokaler Variable oder ohne - direkte Berechnung und Rückgabe des Ergebnisses an
   die aufrufende Funktion
 */
console.log("function triangleArea with side ", s, " & height ", h, "=", triangleArea(s, h));
```

Teilauftrag 4: Dreiecksumfang

```
const pi = 3.14; //accuracy is sufficient enough for our purpose
var sA=3, sB=10, sC=5;

function triangleCircumfence(sideA, sideB, sideC) {
    let cf = sideA+sideB+sideC; // lokale Variable, nur innerhalb Funktion sichtbar
    return cf;
}

console.log("function triangleCircumfence with sideA:", sA, ", sideB:", sB,
    " sC:", sC, "=",triangleCircumfence(sA, sB, sC));
```

Teilauftrag 5: quadratFlaeche

```
const pi = 3.14; //accuracy is sufficient enough for our purpose
var side = 20;

console.log("function square: ", side, " = ", squareArea(side));

function squareArea(x) {
    let sa = x*x;
    return sa;
}
```

Teilauftrag 6: quadratUmfang

```
const pi = 3.14; //accuracy is sufficient enough for our purpose
var side = 4;

function squareCircumference(s) {
    let cf = 4*s; // lokale Variablen
    return cf;
}

console.log();
console.log("function squareCircumference: side ", side +" =
",squareCircumference (side));
```

Teilauftrag 7: rechtEckFlaeche

```
const pi = 3.14; //accuracy is sufficient enough for our purpose
var sideA = 10, sideB=15;

function rectangleArea(sA, sB) {
    let rA = sA*sB;
    return rA;
}

console.log();
console.log("function rectangleArea: sideA:", sideA, " sideB:", sideB, " =
",
    rectangleArea(sideA, sideB));
```

Teilauftrag 8: rechteckUmfang

```
const pi = 3.14; //accuracy is sufficient enough for our purpose
var sideA = 10, sideB=15;

function rectangleCurcumference(sA, sB) {
    let rcf = 2*(sA+sB);
    return rcf;
}

console.log();
console.log("function rectangleCurcumfence: sideA:", sideA, " sideB:",
sideB,
    " = ", rectangleCurcumference(sideA, sideB));
```

Volkan Demir

From:
https://wiki.bzz.ch/ - **BZZ - Modulwiki**

Permanent link:
https://wiki.bzz.ch/de/modul/m307/learningunits/lu01/loesungen/02

Last update: **2026/04/07 08:58**