

Name:

Matrikelnummer:

Webapplikationen
SS 2017
26.07.2017

Name:

Vorname:

Matrikelnummer:

Semester:

Schreiben Sie unbedingt auf jedes Blatt Namen, Vornamen und Matrikelnummer!

Aufgabe	1	2	3	4	5	6	7
Punkte							

Summe (max. 90 Punkte):

Note:

Viel Erfolg!

Name:

Matrikelnummer:

1. Aufgabe (Konzepte)

10 Punkte

Nennen Sie vier wesentliche Unterschiede zwischen Java und JavaScript!

Erklären Sie den Begriff der „Callback-Funktion“! Nennen Sie (im passenden Kontext) ein Beispiel einer Callback-Funktion!

Was ist die Laufzeitumgebung eines

- Java-Programms:
- JavaScript-Programms:
- PHP-Programms:

Name:

Matrikelnummer:

2. Aufgabe (Objekte)**15 Punkte**

a) Finden und verbessern Sie mindestens 18 Fehler im nachfolgenden JavaScript-Code!

```
var Figur = funktion(var id) {
    this._id = id;
};

Figur.prototype.berechneFlaecheninhalt = function() {};
Figur.prototype.berechneUmfang = function() {};

Figur.holeId = function() {
    return _id;
};

Rechteck = function(id, l, t, w, h) {
    Figur.call(id);
    this._left = l;
    this._top = t;
    this._width = w;
    this._height = h;
};

Rechteck.prototype = Object.create(Figur);
Rechteck.prototype = Rechteck;

Rechteck.prototype.berechneFlaecheninhalt = function() {
    return _width * _height;
};

Rechteck.prototype.berechneUmfang = function() {
    return "2" * (this._width + this._height)
}

var Kreis = function(id, x, y, r)
    Figur (this, id);
    this._cx = x;
    this._cy = y;
    this._radius := r;
};

Kreis.prototype = Object.create(Figur.prototype);
Kreis.prototype.constructor = Rechteck;

Kreis.prototype.berechneFlaecheninhalt = function {
    return Math.PI * this._radius * this._radius
};

Kreis.prototype.berechneUmfang = Function() {
    2 * Math.PI * _radius;
};

var r = new Rechteck("R", 150, 50);
k = new Kreis("K", 50, 50, 20);

console.log("Umfang Rechteck: " . r.berechneUmfang().toFixed(2));
console.log("Fläche Kreis: " + k.berechneFlaecheninhalt.toFixed(2))
```

Name:

Matrikelnummer:

b) Stellen Sie folgende Rechnung als JSON-Objekt dar!

Rechnung

Herrn Erich Mustermann
Mustertrasse 1
12345 Musterstadt

Menge	Artikelbezeichnung	Einzelpreis	Gesamt
2	GTX1016	250	500
1	Core i7 6400	300	300
Summe:			800

Name:

Matrikelnummer:

3. Aufgabe (Grundlegende Programmierkenntnisse in JavaScript) 10 Punkte

Schreiben Sie eine JavaScript-Funktion, welche für ein Eingabearray mit Zahlenwerten `true` zurückliefert, wenn auf jede 7 im Array direkt danach mindestens eine weitere 7 folgt!

Beispiel:

```
folgt7([1, 7, 7, 1]) → true  
folgt7 ([7, 1, 7]) → false  
folgt7 ([1, 3, 7, 7, 4, 7, 7]) → true  
folgt7 ([1, 3]) → false  
folgt7 ([]) → false  
folgt7 ([8, 7, 7, 7, 5]) → true
```

Lösung:

```
function folgt7 (
```

```
) {
```

```
}
```

Name:

Matrikelnummer:

4. Aufgabe (PHP)

10 Punkte

Schreiben Sie ein PHP-Skript `encryptString.php` zum Verschlüsseln eines übergebenen Strings. Der Aufruf erfolgt im Browser z.B. über

`http://localhost:8080/encryptString.php?someString=Mustermann`

Die Funktion liefert als Ergebnis den Originalstring gefolgt von einem '_'-Zeichen gefolgt von dem verschlüsselten String zurück, z.B. `Mustermann_x07uzV-T(b`

Sie dürfen für die Aufgabe davon ausgehen, dass eine PHP-Hilfsfunktion `encrypt()` zum eigentlichen Verschlüsseln gegeben ist, die Sie in Ihrem PHP-Skript verwenden können.

Name:

Matrikelnummer:

5. Aufgabe (DOM)

15 Punkte

a) Zeichnen Sie für folgende HTML Seite den zugehörigen DOM-Tree:

```
<html>
<head>
<title>Good Luck!</title>
<meta charset="UTF-8">
</head>
<body>
<script>
alert("Hello World");
</script>
<form id="myForm" action="action_page.php">
Name: <input type="text" name="fname">
<br><input type="submit" value="Submit">
</form>
</body>
</html>
```

b) Wenn diese Seite geladen wird, erscheinen dann zuerst die Formular-Elemente ☐ oder kommt zuerst die Alert-Box ☐ ? Bitte ankreuzen!

c) Fügen Sie im `<script>`-Tag zusätzlich per JavaScript nach dem `Submit`-Button einen `Reset`-Button ein. (Tipp: Beachten Sie hier Aufgabenteil b.)

Name:

Matrikelnummer:

6. Aufgabe (XMLHttpRequest)

20 Punkte

In dieser Aufgabe soll mittels einer HTML-Seite, welche Teil eines Einkaufsprozesses ist, von einem Benutzer dessen Name sowie dessen Kreditkartennummer an das PHP-Skript `verifyCard.php` auf dem gleichen Server gesendet werden, welches die Gültigkeit der Kreditkarte prüft.

Dabei wird beim Verlassen (`blur`-Event) des Eingabefeldes namens `kreditkartennr` über AJAX besagtes PHP-Skript aufgerufen. Nur wenn die Prüfung erfolgreich ist, wird der Button `Fortsetzen` aktiviert.

Das PHP-Skript `verifyCard.php` sei gegeben und muss von Ihnen **NICHT** programmiert werden! Programmieren Sie lediglich den entsprechenden JavaScript-Teil und notieren Sie die dafür relevanten HTML-Elemente.

Hinweis 1: Sie können einen Button aktivieren, indem Sie dessen `disabled` Attribut auf `false` setzen.

Hinweis 2: Sie dürfen (für die Aufgabe) eine ideale Welt ohne böse Menschen annehmen und GET verwenden.



Kreditkartenpruefung

(AJAX Version)

Name:

Kreditkartennummer:

HTML-Code zur Darstellung der Seite:

Name:

Matrikelnummer:

6. Aufgabe (Fortsetzung, JS-Code)

Name:

Matrikelnummer:

7. Aufgabe (Rekursion)

10 Punkte

Ein Bauer in der Rhön möchte wissen, wie viele Ohren die Schafe in seiner Herde haben. Schreiben Sie in JavaScript eine **rekursive** Funktion `schafsOhren(anzahlSchafe)`, die **ohne** Verwendung von Hilfsvariablen, Schleifen und Multiplikation die Anzahl der Ohren aller Schafe berechnet!

Beachte: Für eine triviale Lösung wie z.B. `function schafsOhren(anzSchafe) { return 2 * anzSchafe; }` gibt es keine Punkte!

Lösung:

```
function schafsOhren(                ) {
```

```
}
```