**HTML & CSS**

**1) *Nennen Sie die beiden Tags, die zwei wichtige Bereiche des HTML unterscheiden, und erläutern Sie kurz die üblichen Inhalte dieser Tags.***

- Antwort: Die beiden wichtigen Tags sind `<head>` und `<body>`.

- Im `<head>`-Tag werden Metadaten für das Dokument gespeichert, wie der Titel der Seite (`<title>`), Links zu Stylesheets und Skripten sowie Meta-Beschreibungen.

- Im `<body>`-Tag sind die sichtbaren Inhalte der Seite enthalten, wie Texte, Bilder und Links.

***4) Nennen Sie drei verschiedene Selektorarten ohne Kombinatoren für CSS-Regeln.***

- Antwort: Elementselektor, Klassenselektor, ID-Selektor.

***5) Nenne zwei Input-Tag-Typen, aber nicht Texteingabe oder Schaltfläche.***

- Antwort: `<input type="checkbox">`, `<input type="radio">`.

***7) Nenne drei CSS-Stile, keine Regeln und Selektoren.***

- Antwort: `color`, `font-size`, `margin`.

***8a) Nenne drei CSS-Selektorarten.***

- Antwort: Element-, Klassen- und ID-Selektoren.

***8b) Nenne zwei Nachbarselektoren.***

- Antwort: Adjacent Sibling Selector (`+`), Descendant Selector (` `).

***x) Nenne zwei Möglichkeiten, CSS-Selektoren zu kombinieren.***

- Antwort: Kombinieren durch Nachbarschaft (`element1 + element2`), Verschachtelung (Descendant Selector: `element1 element2`).

***y) Darf ein Artikel-Tag `section`-Tags enthalten?***

- Antwort: Ja, `<article>`-Tags können `<section>`-Tags enthalten, wenn es der inhaltlichen Struktur dient.

***z) Nenne je ein Block-Tag und ein Inline-Tag.***

- Antwort: Block-Tag: `<div>`, Inline-Tag: `<span>`.

***w) Zwei Tags aus dem Bereich klassisches syntaktisches HTML***

- Antwort: `<b>`, `<i>`.

***v) Zwei Tags aus dem Bereich semantisches HTML.***

- Antwort: `<header>`, `<footer>`.

**JavaScript**

***13) In welcher Software wird das DOM genutzt?***

- Antwort: Das DOM (Document Object Model) wird hauptsächlich in Webbrowsern genutzt, die HTML- und JavaScript-Inhalte verarbeiten.

***14a) Schreiben Sie eine Definition der Variablen `int`, die an einen Ganzzahlenwert gebunden ist.***

- Antwort: `let int = 42;` (JavaScript).

***14b) Schreiben Sie eine Funktion, die die Summe zweier Zahlen berechnet und zurückgibt. Rufen Sie diese Funktion nach der Definition auf.***

- Antwort:

function sum(a, b) {

return a + b;

}

let result = sum(3, 5);

***16) Wie lautet das JavaScript-Kodefragment, um das erste DOM-Objekt zu finden, das die Klasse „standardtext“ hat?***

- Antwort: `document.querySelector(".standardtext");`

***17) Schreiben Sie das JavaScript-Kodefragment, das dem Absatz mit der Id „Ziel“ die CSS-Klasse `.huebsch` hinzufügt.***

- Antwort: `document.getElementById("Ziel").classList.add("huebsch");`

***25) Schreiben Sie ein Codefragment, das für alle mit der Funktionsbezeichnung „Auszubildender“ eine Ausgabe erzeugt und in die Webseite einfügt.***

- Antwort:

let personen = { "Müller": "Auszubildender", "Schmidt": "Manager" };

for (let nachname in personen) {

if (personen[nachname] === "Auszubildender") {

document.body.innerHTML += `<p>${nachname}: ${personen[nachname]}</p>`;

}

}

***26) Schreiben Sie den Ausdruck für das Parsen eines JSON-Strings in JavaScript.***

- Antwort: `let obj = JSON.parse(jsonString);`

***29) Validierung durch den Client - muss der Server dies noch einmal machen?***

- Antwort: Trifft teilweise zu. Der Client sollte vorvalidieren, aber der Server muss alle Daten erneut prüfen, um Manipulation zu verhindern.

***16) HTML-Fragment für einen Link, der die Funktion `esfunzt` bei einem Klick aufruft.***

- Antwort:

```html

<a href="#" onclick="esfunzt()">Klicken Sie hier</a>

```

**PHP**

***23) Schreiben Sie unter hier einen Wert für die Variable X, so dass die beiden folgenden Expressions `true` liefern.***

- Antwort:

Um die Bedingung zu erfüllen, dass sowohl $x == 88 als auch $x == "88" true ergibt, können Sie den Wert von $x als Zeichenkette "88" setzen. In PHP werden beim Vergleich mit == (lose comparison) Typen automatisch konvertiert, sodass sowohl ein numerischer Vergleich (88 == "88") als auch ein stringbasierter Vergleich ("88" == "88") true ergibt.

```php

Also: $x = „88“;

```

***26) Schreiben Sie jeweils den Ausdruck hin, um einen JSON-String in PHP zu dekodieren.***

- Antwort: `$obj = json\_decode($jsonString, true);`

**Reguläre Ausdrücke(?)**

***27) Regulärer Ausdruck für Vor- und Nachnamen-Kombinationen:***

- Antwort:

```regex

^[A-Z][a-z]{1,15} [A-Z][a-z]{1,15}(-[A-Z][a-z]{1,15})?$

```

A) Genau ein Leerzeichen: `\s`

B) Ein Namensteil: `[A-Z][a-z]{1,15}`

C) Kombination (Gesamtergebnis): `^[A-Z][a-z]{1,15} [A-Z][a-z]{1,15}(-[A-Z][a-z]{1,15})?$`

**Webserver und REST**

***21a) Serverroot und Documentenroot auf einem Webserver - was findet man dort?***

- Antwort:

- Serverroot: Enthält Konfigurationsdateien und systemweite Einstellungen.

- Documentroot: Speichert Inhalte, die der Webserver ausliefert.

***22) Eine alternative Möglichkeit zur Datenpersistenz auf einem Webserver.***

- Antwort: Lokale Dateien, beispielsweise JSON- oder XML-Dateien, um Daten zu speichern.

***c) Wie kann man herausfinden, ob das PHP-Modul auf dem Apache-Webserver aktiv ist?***

- Antwort: Eine Datei mit dem Inhalt `<?php phpinfo(); ?>` erstellen und im Browser öffnen. Wenn PHP aktiviert ist, werden PHP-Konfigurationsinformationen angezeigt.