**HTML:**

- div, span, p, címsorok

- képek

- táblázat

- linkek

- - űrlap elemek:

- **szöveges beviteli mező: egy**, több soros

- checkbox

- radio button

- színválasztó

- dátum választás

- gombok

- számozott lista

- pontozott lista

- több darab html fájl legyen, minimum 4

**CSS:**

- style attribútumban néhány elem formázása

- azonosító alapján formázást

- osztály alapján formázást

- táblázat formázása

- menü kialakítása

- háttérszín

- linkek formázása

- űrlap elemek, gombok formázása

- egy vagy több css fájlban is legyen formázás, és a html-ben a <style> tag-ben is

**Javascript, jQuery:**

- form ellenőrzés, hiba esetén az egyes űrlap elemeknél kiírja hogy mi a hiba, css-t állít pl. piros keret ha hibás az adott űrlap elem

- animáció

- elemek kiválasztása: html tag név alapján, osztály alapján, azonosító (id) alapján stb.

- új html elem készítése, meglévő html elem módosítása

Dokumentáció is szükséges az elvégzett munkáról (a kód és felhasználói felület bemutatása). Minimum 5 oldal, nincs maximum korlát.

# A feladat leírása

A beadandó témája egy weboldal létrehozása HTML nyelven, a megadott feltételek alapján. A létrehozott weboldal egy modern hotelweb alkalmazás, melyet úgy terveztem és fejlesztettem ki, hogy a felhasználók számára könnyű és könnyen kezelhető élményt nyújtson a hotel szolgáltatásainak böngészése és online foglalása során.

A beadandó több HTML fájlt is tartalmaz, amelyekhez közös CSS fájl tartozik, de a feladat követelményeinek megfelelve néhol a HTML fájlban style tagek között, illetve attribútumként szerepel a formázás. Igyekeztem esztétikus, letisztult dizájnra törekedni A weboldal menüszerkezete a főoldal mellett tartalmazza a Szobákat, Rólunkat, Szállodáinkat és Kapcsolat menüpontokat, biztosítva a könnyű navigációt. A látványkép és a színpaletta a hotel eleganciáját és kényelmét tükrözi.

Tartalmaz még egy javascript fájlt is, amely felelős a formban szereplő adatok validálásáért, akciókért és egyebekért.

# HTML fájlok

## home.html

## rooms.html

## about.html

## ourHotels.html

## contact.html

# CSS fájl

## style.css

# JavaScript fájl

## javaScript.js

# HTML követelmények:

# CSS követelmények:

## 1. Style attribútumban néhány elem formázása:

- Példa: <div style="width: 100% !important; margin-left: 10px;"> elem formázása az attribútumnál.

## 2. Azonosító alapján formázás:

- Példa: `#section2 ` kiválasztása és formázása.

#section2 {

display: flex;

width: 100%;

justify-content: space-between;

}

## 3. Osztály alapján formázás:

- Példa: `.fekete` osztályú gomb formázása.

.fekete {

border: none;

background-color: black;

color: white;

}

## 4. Táblázat formázása:

- Példa: `table` és `th`, `td` elemek formázása a táblázatban.

table {

width: 100%;

border-collapse: collapse;

margin-top: 20px;

}

th {

background-color: #f2f2f2;

color: #333;

font-weight: bold;

padding: 10px;

text-align: left;

}

td {

border: 1px solid #ddd;

padding: 10px;

}

## 5. Menü kialakítása:

- Példa: `nav` és `nav a` elemek formázása a menüben.

<nav>

<a href="home.html" class="logo"><i class="fa fa-bed"></i> Logo</a>

<a href="rooms.html">Rooms</a>

<a href="about.html">About</a>

<a href="ourHotels.html">Our Hotels</a>

<a href="contact.html">Contact</a>

<a href="contact.html" class="booknow">Book Now</a>

</nav>

body > nav {

color: #000;

background-color: #fff;

font-size: 18px;

width: 100%;

overflow: hidden;

display: flex;

}

body > nav a {

text-decoration: none;

width: auto;

padding: 8px 16px;

color: #000;

text-align: center;

line-height: 1.5;

}

## 6. Háttérszín:

- Példa: `body` és `body > footer` háttérszínének beállítása.

background-color: #fff;

## 7. Linkek formázása:

- Példa: `body > nav a:hover` és `.linkek a:hover` linkek hover állapotban történő formázása.

body > nav a:hover {

background-color: #aaa;

}

.linkek a:hover {

color: #aaa;

}

## 8. Űrlap elemek, gombok formázása:

- Példa: `main > section:first-child > div > div > div > input` és `.kereso\_gomb ` űrlap és gomb formázása.

body > article > form > div > div {

width: 47%;

padding: 16px 0 20px;

}

.kereso\_gomb {

font-size: 15px;

color: #fff;

background-color: #616161;

padding: 10px 16px;

margin: 10px 16px 20px;

border: 0;

outline: 0;

}

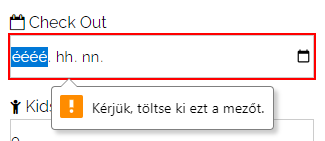
## 9. CSS fájlban formázás és a html-ben a `<style>` tag-ben is:

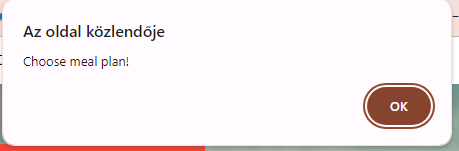
- Példa: A stílusokat a külső CSS fájlban (`style.css`) és az inline stílusokat a HTML `<style>` tag-ben egyaránt megtalálhatók.

# JS követelmények:

## form ellenőrzés

Ez a JavaScript kód egy űrlap validálást végző funkciót tartalmaz, mely ellenőrzi a felhasználó által kitöltött adatok helyességét. Ha a dátumok formátuma vagy a megadott személyek száma nem megfelelő, vagy ha nincs kiválasztva étkezési terv, akkor hibaüzenetek jelennek meg. Ha az adatok helyesek, akkor a kód egyéb része meghívódik, és létrehoz egy szövegfájlt (booking\_data.txt), mely tartalmazza a beírt adatokat, mint például a bejelentkezés és kijelentkezés dátumát, felnőttek számát, kiválasztott étkezési tervet, valamint a szobaszín kódját. A fájl automatikusan letöltődik, így a felhasználó elmentheti a foglalási adatokat a számítógépére.





Funkciókat megvalósító kód:

function validateForm() {

var checkInDate = document.getElementById('checkInDate').value;

var checkOutDate = document.getElementById('checkOutDate').value;

var adults = document.getElementById('adults').value;

var breakfast = document.querySelector('input[name="mealPlan"]:checked');

var roomColor = document.getElementById('roomColor').value;

if (!isValidDate(checkInDate) || !isValidDate(checkOutDate)) {

showError("Wrong date format");

return false;

}

if (adults < 1) {

showError("Minimum 1 person");

return false;

}

if (!breakfast) {

showError("Choose meal plan!");

return false;

}

// Mentés a fájlba

saveDataToFile(checkInDate, checkOutDate, adults, breakfast.value, roomColor);

return true;

}

function showError(message) {

alert(message);

}

function isValidDate(dateString) {

var regex = /^\d{4}-\d{2}-\d{2}$/;

return dateString.match(regex) !== null;

}

function saveDataToFile(checkInDate, checkOutDate, adults, mealPlan, roomColor) {

var dataToSave = `Check In Date: ${checkInDate}\nCheck Out Date: ${checkOutDate}\nAdults: ${adults}\nMeal Plan: ${mealPlan}\nRoom Color: ${roomColor}`;

var blob = new Blob([dataToSave], { type: 'text/plain' });

var a = document.createElement('a');

a.href = window.URL.createObjectURL(blob);

a.download = 'booking\_data.txt';

document.body.appendChild(a);

a.click();

document.body.removeChild(a);

}

## animáció

## elemek kiválasztása: html tag név alapján, osztály alapján, azonosító (id) alapján stb.

A JavaScript kódban található különböző elemek kiválasztására szolgáló módszerek lehetővé teszik az HTML dokumentumon belüli elemek célzását és manipulációját. Az alábbi példákban látható, hogyan lehet kiválasztani elemeket HTML tag név, osztály és azonosító (id) alapján.

példák:

var roomColor = document.getElementById('roomColor').value;

var elementsWithClass = document.getElementsByClassName('bevitel1');

var allInputs = document.getElementsByTagName('input');

## új html elem készítése, meglévő html elem módosítása