

# Web programozás II. – Előrehozott ZH

## 1. AJAX feladatok

### 1.2 XML tartalom feldolgozás

A feladat megoldásához tölts le a **food\_menu.xml** fájlt!

Készítsen egy egyszerű *html* oldalt. Az oldalon legyen **egy gomb** és **egy HTML elem** (pl. *div*) az eredmények megjelenítéséhez.

A feladat egy AJAX kommunikáció megvalósítása aszinkron GET kéréssel. A kérés a gombra kattintás után küldődjön el, majd a feldolgozott válasz tartalma jelenjen meg az eredmények megjelenítésére létrehozott HTML elemen belül. Az adatok lekérése a **food\_menu.xml** állományból történjen.

A **gomb** megnyomását követően a **HTML elem**ben jelenjen meg a „Belgian” szót tartalmazó ételek (food) neve (name) ABC sorrendbe rendezve.

Elvárt kimenet:

Belgian Waffles, Berry-Berry Belgian Waffles, Strawberry Belgian Waffles
--

### 1.2 JSON tartalom feldolgozás

Égészítse ki az előző feladatban létrehozott *html* oldalt. Az oldalon legyen (még) **egy gomb** és **egy HTML elem** az eredmények megjelenítéséhez.

A feladat egy AJAX kommunikáció megvalósítása aszinkron GET kéréssel. A kérés a gombra kattintás után küldődjön el, majd a feldolgozott válasz tartalma jelenjen meg az eredmények megjelenítésére létrehozott elemen belül. Az adatok lekérése a következő API végpontról történjen:

<https://restcountries.com/v3.1/all>

A **gomb** megnyomását követően a **HTML elem**ben jelenjen meg az összes olyan ország hivatalos nevének (official) magyar fordítása (hun), amely Afrikában van (continents). Az országok neveit rendezetlen listában (ul) jelenítse meg.

Elvárt kimenet:

- |  |
|--|
| <ul style="list-style-type: none"><li>• Sierra Leone Köztársaság</li><li>• Elefántcsontparti Köztársaság</li><li>• Tunéziai Köztársaság</li><li>• Benini Köztársaság</li><li>• Zöld-foki Köztársaság</li><li>• Ugandai Köztársaság</li><li>...</li></ul> |
|--|

## 2. Node.js feladatok

A feladat egy egyszerű **Node.js** **szerver** megvalósítása az **Express** keretrendszer használatával. A szerver a bejövő kérésekre a **2501-es porton** figyeljen.

### 2.1 URL feldolgozás

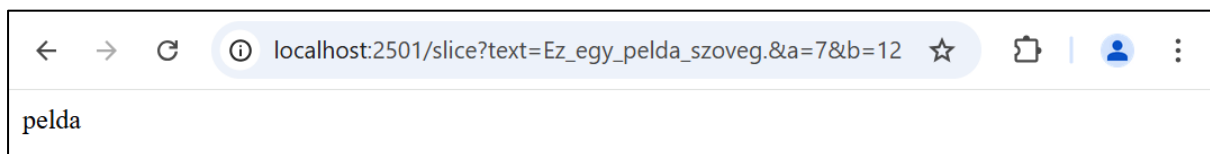
A szerver **dolgozzon fel** egy olyan **URL-t**, ami egy szöveget (**text**) és két értéket tartalmaz (**a** és **b**) query string formában. Például, ha a szöveg „Ez\_egy\_pelda\_szoveg.” és a két szám 7 és 12, akkor az URL a következő:

[http://localhost:2501/slice?text=Ez\\_egy\\_pelda\\_szoveg.&a=7&b=12](http://localhost:2501/slice?text=Ez_egy_pelda_szoveg.&a=7&b=12)

Egészítse ki a szerver kódját úgy, hogy ha a kérés útvonalában a **„/slice”** szerepel, akkor történjen az URL feldolgozás. A **text** értékeként megadott szövegre hívja meg a slice() metódust, aminek a paraméter értékeit **a** és **b** határozza meg.

- Ha mindegyik paraméter meg van adva, akkor a válaszüzenet a metódus kimenete legyen. Az **a** és **b** értéke is nem negatív egész szám. A **b** értéke nem lehet kisebb az **a** értékénél és nem lehet nagyobb a **text** hosszánál.
- Ellenkező esetben a következő szöveg jelenjen meg: *„Kérem, adja meg az összes paramétert!”*

Példa a működésre:



### 2.2 Modul készítése és használata

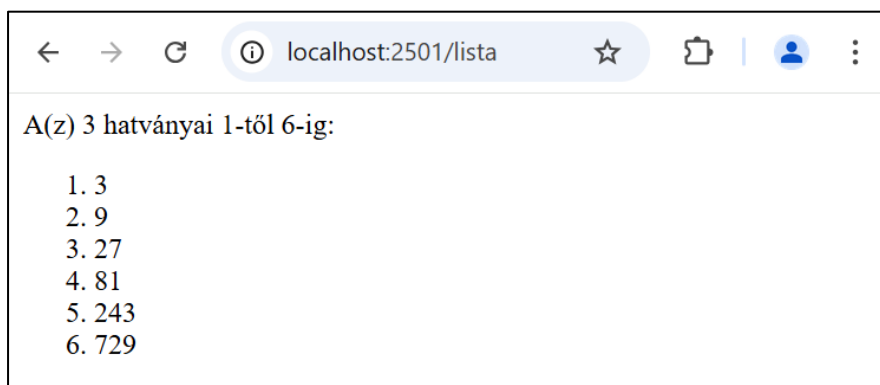
Egy külön fájlban készítsen egy **modult**, ami egy olyan függvényt tartalmaz, ami **dinamikusan generál egy rendezett listát** (ol). A függvénynek két paramétere legyen:

- **n**: hatványalap (pozitív egész szám), és
- **k**: a kitevő maximális értéke (pozitív egész szám).

A függvény az **n** számot 1-től **k**-ig hatványozva építi fel a listát.

A szerver kódjába importálja a modult és egészítse ki a szerver kódját úgy, hogy ha a kérés útvonalában a **„/lista”** szerepel, akkor ez a függvény kerüljön meghívásra a válaszüzenetben. A megadott paraméter értékek **véletlenszerűen generált, 2 és 10 közötti egész számok** legyenek. A generálás külön-külön történjen a két paraméterhez minden kéréskor.

Példa a működésre:



## További információk

A megoldáshoz használhatók saját jegyzetek és kódok, valamint a W3Schools weboldala.

A kurzus Moodle oldaláról letölthető és használható: korábbi órai kódok & XAMPP segédlet.

**Hallgatótársak és AI chatbotok (pl. ChatGPT) használata szigorúan tilos!**

**A feladat végeztével csomagolja össze fájljait egy ZIP állományba és töltsse fel a Moodle-be az erre kijelölt feltöltő felületen! A ZIP fájl neve legyen a Neptun kódja!**

Az állomány lehetséges tartalma:

- AJAX
  - index.html
  - scripts.js
  - plant\_catalog.xml
- NodeJS
  - app.js
  - modul.js
  - egyéb fájlok (package.json, package-lock.json)

**Jó munkát!**