# Backend gyakorló feladatok

A feladat célja, hogy a hallgatók REST API elérési pontokat hozzanak létre egy programozási nyelvben és annak keretrendszerében. A feladatokhoz megadott CSV fájlok tartalmazzák a forrásadatokat. A feladatok megoldásához az SQL lekérdezéseket is meg kell adni, valamint a várható válaszokat a megadott adathalmaz alapján.

**1. Feladat** Készíts egy REST API végpontot, amely visszaadja az események számát.

* **Elérési pont:** GET /api/events/count
* **SQL:** SELECT COUNT(\*) AS event\_count FROM event;
* **Várható válasz:** { "event\_count": 25 }

**2. Feladat** Készíts egy REST API végpontot, amely visszaadja a legkorábbi esemény dátumát.

* **Elérési pont:** GET /api/events/min\_date
* **SQL:** SELECT MIN(EventDateTime) AS earliest\_event FROM event;
* **Várható válasz:** { "earliest\_event": "2023-08-01 18:00:00" }

**3. Feladat** Készíts egy REST API végpontot, amely visszaadja azokat az eseményeket, amelyek 2025 után vannak ÉS "konferencia" szerepel a címükben.

* **Elérési pont:** GET /api/events/future\_conferences
* **SQL:** SELECT \* FROM event WHERE EventDateTime > '2025-01-01' AND Title LIKE '%konferencia%';
* **Várható válasz:** { "events": [{ "ID": 16, "Title": "Online marketing konferencia", "EventDateTime": "2025-08-12 09:00:00" }, { "ID": 24, "Title": "Digitális jövő konferencia", "EventDateTime": "2043-06-30 11:30:00" }] }

**4. Feladat** Készíts egy REST API végpontot, amely visszaadja az eseményeket csak a cím és az időpont oszlopokkal, időrendben rendezve.

* **Elérési pont:** GET /api/events/titles\_sorted
* **SQL:** SELECT Title, EventDateTime FROM event ORDER BY EventDateTime ASC;
* **Várható válasz:** { "events": [{ "Title": "Nyári fesztivál", "EventDateTime": "2023-08-01 18:00:00" }, ...] }

**5. Feladat** Készíts egy REST API végpontot, amely visszaadja, hogy hány eseményt szervezett egy-egy szerző.

* **Elérési pont:** GET /api/authors/event\_count
* **SQL:** SELECT AuthorID, COUNT(\*) AS event\_count FROM event GROUP BY AuthorID;
* **Várható válasz:** { "author\_events": [{ "AuthorID": 1, "event\_count": 3 }, { "AuthorID": 2, "event\_count": 3 }, ...] }

**6. Feladat** Készíts egy REST API végpontot, amely visszaadja az adott szerző eseményeit.

* **Elérési pont:** GET /api/events/by\_author/{author\_id}
* **SQL:** SELECT \* FROM event WHERE AuthorID = ?;
* **Paramétere(k):** author\_id (pl. 1)
* **Várható válasz:** { "events": [{ "ID": 1, "Title": "Nyári fesztivál", "EventDateTime": "2023-08-01 18:00:00" }] }

**7. Feladat** Készíts egy REST API végpontot, amely lehetővé teszi egy esemény címének módosítását ID alapján.

* **Elérési pont:** PUT /api/events/update\_title
* **SQL:** UPDATE event SET Title = ? WHERE ID = ?;
* **Paramétere(k):** ID, Title
* **Várható válasz:** { "success": true }

**8. Feladat** Készíts egy REST API végpontot, amely lehetővé teszi egy esemény törlését ID alapján.

* **Elérési pont:** DELETE /api/events/delete/{id}
* **SQL:** DELETE FROM event WHERE ID = ?;
* **Paramétere(k):** id (pl. 1)
* **Várható válasz:** { "success": true }

**9. Feladat** Készíts egy REST API végpontot, amely visszaadja az eseményeket és azok szerzőjét.

* **Elérési pont:** GET /api/events\_with\_authors
* **SQL:** SELECT event.\*, author.Name FROM event INNER JOIN author ON event.AuthorID = author.ID;
* **Paramétere(k):** Nincs
* **Várható válasz:** { "events": [{ "ID": 1, "Title": "Nyári fesztivál", "Author": "Kiss Péter" }] }