**BACKEND PROGRAMOZÁS**

**TANFOLYAM feladat**

1. Hozzon létre backend szerver projektet az Ön által választott programnyelven, illetve fejlesztési környezetben! A projektmappát 2025\_03\_04 néven nevezze el!
2. Hozzon létre a lokális SQL szerveren **tanfolyam** néven adatbázist! Az adatbázis alapértelmezett rendezési sorrendje a magyar szabályok szerinti legyen! Állítsa be az UTF-8 kódolást alapértelmezettnek az adatbázis létrehozásánál! Az SQL mondatot másolja a *lekerdezesek.sql* állományba (2. feladat:)

Az *tanfolyam.sql* állomány tartalmazza a táblákat létrehozó, és az adatokat a táblába beszúró SQL parancsokat! Futtassa a parancsfájlt az *tanfolyam* adatbázisban!

***Adatbázis szerkezete:***

1. tanulok **(Tanulók tábla)**

Tárolja a tanulók alapadatait:

* + id (INT) – Egyedi azonosító
  + nev (VARCHAR) – Tanuló neve
  + telefonszam (VARCHAR) – Tanuló telefonszáma
  + szuletesido (DATE) – Tanuló születési dátuma
  + email (VARCHAR) – Tanuló e-mail címe

1. ertekelesek **(Értékelések tábla)**

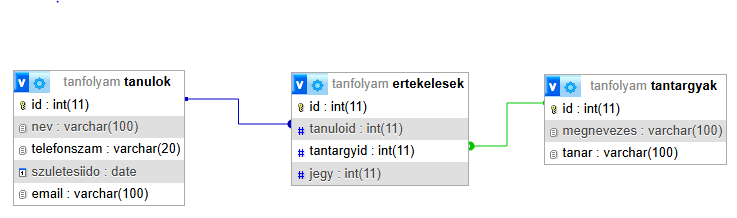
Összekapcsolja a tanulókat a tantárgyakkal és a kapott jegyekkel:

* + id (INT) – Egyedi azonosító
  + tanuloid (INT) – Kapcsolat a tanulók táblával
  + tantargyid (INT) – Kapcsolat a tantárgyak táblával
  + jegy (INT) – Tanuló által kapott jegy

1. tantargyak **(Tantárgyak tábla)**

Tárolja a tantárgyakat és az oktatókat:

* + id (INT) – Egyedi azonosító
  + megnevezes (VARCHAR) – Tantárgy neve
  + tanar (VARCHAR) – Tantárgyat oktató tanár neve



1. Hozzon létre egy **PHP alapú API-t**, amely ***JSON formátumban visszaadja az összes tanuló adatát*** az adatbázisból. Az **API** működése:
2. Csatlakozik az adatbázishoz
3. Az API lekérdezi az összes tanuló ***nevét, születési dátumát és email címét*** az adatbázisból.
4. **JSON** formátumban visszaadja az adatokat

|  |  |  |  |
| --- | --- | --- | --- |
| Metódus | URL | Body | Válasz |
| **GET** | **/api/tanulok.php** | **üres** | **JSON** |

Válasz az adott mintának legyen megfelelő:

[

{

"id": 1,

"nev": "Kovács Péter",

"telefonszam": "+36-70-1234567",

"szuletesiido": "2000-05-14",

"email": "peter.kovacs@example.com"

},

…

]

1. Készítsen egy **PHP API végpontot**, amely egy adott ***tantárgy neve* alapján** visszaadja a tanulói ***átlagát*** az adott tantárgyból. Az **API**:
2. Kliens küldi a kérést a "tantargy" paraméterrel.
3. Az **API** ellenőrzi, hogy van-e ilyen tantárgy a tantargyak táblában, ha nem, akkor hibaüzenetet küld.
4. Az SQL lekérdezés:

* Összekapcsolja az ertekelesek, tanulok és tantargyak táblákat.
* Kiválasztja a tanulók nevét és az adott tantárgyból szerzett jegyeiket.
* Kiszámítja tanulóként az átlagos osztályzatot.
* Biztosítja, hogy minden tanulóhoz külön számolja ki az átlagot.
* Ha egy tanulónak **nincs jegye**, akkor ő nem jelenik meg a listában

1. Az API visszaadja a JSON választ, amely tartalmazza a tanulók nevét és az átlagukat, vagy hibaüzenetet küld az alábbi mintának megfelelően.

|  |  |  |  |
| --- | --- | --- | --- |
| Metódus | URL | Body | Válasz |
| **POST** | **/api/tanulok.php** | **JSON** | **JSON** |

**Kéréstörzs (Request Body) (JSON):**

{

"tantargy": "Matematika"

}

**Válasz (Response JSON)**

***Sikeres lekérdezés esetén:***

[

{

"tanulo\_nev": "Kovács Péter",

"atlag": 4.5

},

{

"tanulo\_nev": "Nagy Anna",

"atlag": 3.8

},

…

]

***Hiba esetén:***

{

"error": "A megadott tantárgy nem található az adatbázisban."

}

1. Készítsen egy PHP **API** végpontot, amely *új tanulót vesz fel az adatbázisba*. Az **API** működése:

* **A kliens elküldi az *új tanuló adatait*** *JSON* formátumban.
* **Az API ellenőrzi, hogy *minden kötelező mező meg van-e adva (isset)*.**
* **Ha egy mező hiányzik**, hibaüzenetet küld vissza JSON-ben.
* Ellenőrzi, hogy a tanuló adatai nincsenek -e a táblában (email, telefonszam), ha már rögzítve lett, akkor hibaüzenetet küld
* **Ha minden adat helyes**, és nincs még a tanuló a táblában, akkor beszúrja az új tanulót az adatbázisba.
* **JSON válaszban visszaadja a sikeres beszúrás eredményét vagy hibaüzenetet.**

|  |  |  |  |
| --- | --- | --- | --- |
| Metódus | URL | Body | Válasz |
| **POST** | **/api/ujtanulo.php** | **JSON** | **JSON** |

**Kéréstörzs (Request Body) (JSON):**

{

"nev": "Kovács Péter",

"telefonszam": "06201234567",

"szuletesiido": "2005-06-15",

"email": "kovacs.peter@example.com"

}

**Válasz (Response JSON)**

***Sikeres lekérdezés esetén:***

{

"message": "Sikeresen hozzáadtuk a tanulót az adatbázishoz.",

"tanulo": {

"nev": "Kovács Péter",

"telefonszam": "06201234567",

"szuletesiido": "2005-06-15",

"email": "kovacs.peter@example.com"

}

}

***Hiba esetén:***

{

"error": "A megadott e-mail cím vagy telefonszám már létezik."

}

…folytatás!

{

"error": "Minden mezőt ki kell tölteni (nev, telefonszam, szuletesiido, email)."

}

{

"error": "Nem sikerült felvenni a tanulót az adatbázisba!"

}