Táblák részletes magyarázata

1. lockers

* **locker\_id**
* **status**: Jelzi a szekrény aktuális állapotát, ami lehet "nyitott" vagy "zart".
* **can\_be\_opened**: Ez a mező jelzi, hogy a szekrény nyitható-e. Alapértelmezésben hamis, tehát tanári/adminisztrátori jóváhagyás szükséges a nyitáshoz.

1. students

* **student\_id**
* **full\_name**: A diák teljes neve.
* **class**: A diák osztálya.
* **birth\_place**: A diák születési helye.
* **birth\_date**: A diák születési dátuma.
* **rfid\_tag**: A diák RFID azonosítója, amely egyedi és összekapcsolódik a szekrényével.

1. locker\_access\_history

* **access\_id**
* **rfid\_tag**: Az RFID tag, amely a diákhoz van rendelve. Kapcsolódik a students táblához.
* **locker\_id**: A szekrény azonosítója, amelyet a diák ki akar nyitni. Kapcsolódik a lockers táblához.
* **access\_time**: A hozzáférés időpontját rögzíti.
* **approved\_by**: Annak az adminisztrátornak, tanárnak az azonosítója, aki jóváhagyta a hozzáférést. NULL lehet, ha nem szükséges jóváhagyás.

1. locker\_relationships

* **relationship\_id**
* **rfid\_tag**: A diák RFID tag-je. Kapcsolódik a "students" táblához.
* **locker\_id**: A diákhoz rendelt szekrény azonosítója. Kapcsolódik a "lockers" táblához.

Ez a tábla biztosítja, hogy minden diák csak a saját szekrényét tudja kinyitni.

1. subjects

* **subject\_id**
* **subject\_name**: A tantárgy neve.
* **teacher\_name**: A tantárgyat tanító tanár neve.

1. timetables

* **timetable\_id**
* **student\_id**: A diák azonosítója, amely a "students" táblához kapcsolódik.
* **subject\_id**: A tantárgy azonosítója, amely a "subjects" táblához kapcsolódik.
* **day\_of\_week**: Az adott nap az órarendben (hétfő-péntek).
* **start\_time**: Az óra kezdetének időpontja.
* **end\_time**: Az óra végének időpontja.

1. admins

* **admin\_id**: Egyedi azonosító minden adminisztrátor (tanár vagy dolgozó) számára.
* **full\_name**: Az adminisztrátor teljes neve.
* **position**: Az adminisztrátor pozíciója, például "igazgatóhelyettes", "Rendszergazda", "Portás".

Rendszer működése

* A diákok minden reggel beolvastatják az RFID tag-jüket. Ez a rendszer lekérdezi a "locker\_relationships" táblából, hogy melyik szekrényhez tartoznak, majd a "lockers" táblában ellenőrzi, hogy a szekrény nyitható-e.
* Ha a szekrény nyitásához tanári jóváhagyás szükséges, a rendszer ellenőrzi a locker\_access\_history táblát, és rögzíti, hogy ki hagyta jóvá a nyitást.
* Az adminisztrátorok (tanárok vagy dolgozók) az "admins" táblában vannak nyilvántartva, és ők hagyhatják jóvá a hozzáférést egy adott szekrényhez.
* A rendszer minden szekrényhez és diákhoz külön kapcsolódási naplót vezet, biztosítva a hozzáférések nyomon követhetőségét és ellenőrizhetőségét.

1. Kapcsolat az RFID tag és a szekrény között:

Minden RFID tag-et hozzárendelsz egy adott szekrényhez az adatbázisban. Ez a kapcsolat a "locker\_relationships" táblában történik.

Az RFID tag egyedi azonosítót kap, amelyet összekapcsolsz a megfelelő szekrénnyel.

1. Adatbázis logika:

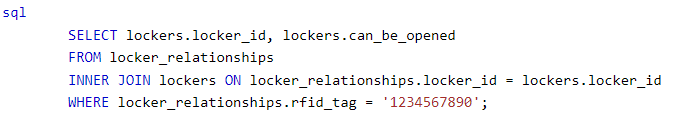
A "locker\_relationships" táblában minden egyes RFID tag egy adott szekrényhez van hozzárendelve. Amikor egy diák beolvassa az RFID tag-et, az adatbázisból lekérdezed, hogy az adott tag melyik szekrényt nyithatja ki.

1. RFID beolvasás logika:

A beolvasott RFID tag azonosítóját elküldöd a szervernek, amely lekérdezi az adatbázisból a megfelelő szekrény azonosítóját (locker\_id).

Ezután ellenőrzöd, hogy az adott szekrény nyitható-e (can\_be\_opened mező), és ha igen, akkor a szerver jelzést küld a szekrény zárjának, hogy nyissa ki.

Példa az adatbázis-lekérdezésre:



Ez a lekérdezés megadja az RFID tag-hez kapcsolódó szekrény azonosítóját és azt is, hogy nyitható-e a szekrény.

1. Zár vezérlése:

Ha a szekrény nyitható, akkor küld egy nyitási parancsot a szekrény zárjának, amely kinyitja a szekrényt.

Folyamat lépései:

* + Diák beolvassa az RFID tag-et.
  + A rendszer lekérdezi az adatbázisból a tag-hez tartozó szekrényt.
  + Ellenőrzi, hogy a szekrény nyitható-e.
  + Ha igen, a zárnyitási parancsot kiadja a megfelelő szekrénynek.

Hardveres megvalósítás:

* + Raspberry Pi, ezen futna a szoftver amely kezli a szekrényeket
  + RFID tag, a diákok a tag-ekket letudják csippantani az RFID leolvasónál
  + RFID leolvasó, beolvassa az RFID tag-eket, és a hozzá rendelt szekrényt kinyitja

Backend:

* + nodeJs
  + Python

Frontend:

* + HTML,CSS
  + JS
  + Bootstrap