

```
C:\xampp\htdocs>composer create-project laravel/laravel F2Ingatlan
Creating a "laravel/laravel" project at "./F2Ingatlan"
Installing laravel/laravel (v11.6.1)
- Installing laravel/laravel (v11.6.1): Extracting archive
Created project in C:\xampp\htdocs\F2Ingatlan
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
Updating dependencies
Lock file operations: 110 installs, 0 updates, 0 removals
- Locking brick/math (0.12.1)
- Locking carbonphp/carbon-doctrine-types (3.2.0)
- Locking dflydev/dot-access-data (v3.0.3)
- Locking doctrine/inflector (2.0.10)
- Locking doctrine/lexer (3.0.1)
- Locking dragonmantank/cron-expression (v3.4.0)
- Locking egulias/email-validator (4.0.3)
- Locking fakerphp/faker (v1.24.1)
- Locking filp/whoops (2.17.0)
```

- 1) Laravel telepítése composer segítségével, utolsó paraméter a Projekt és egyúttal a könyvtár neve is. A parancsot abban a könyvtárban kell kiadni, amelyikben a projektet létre kell hozni.

The screenshot shows the phpMyAdmin interface. On the left is a sidebar with a list of databases. The main area is titled 'Adatbázisok' (Databases). At the top, there's a tabbed interface with 'Adatbázisok' selected. Below the tabs, there's a form to create a new database. The form has a text input for the database name (containing 'adatbazis') and a dropdown for the charset (set to 'utf8mb4_general_ci'). There's a 'Létrehozás' (Create) button. Below the form, there's a table listing existing databases and their collations.

Adatbázis	Illesztés	Művelet
☐ fingatlan	utf8mb4_general_ci	Jogosultságok ellenőrzése
☐ fingatlan2	utf8mb4_general_ci	Jogosultságok ellenőrzése
☐ gitarmuveszek	utf8mb4_general_ci	Jogosultságok ellenőrzése
☐ hamburger	utf8_hungarian_ci	Jogosultságok ellenőrzése
☐ information_schema	utf8_general_ci	Jogosultságok ellenőrzése
☐ ingatlan	utf8mb4_hungarian_ci	Jogosultságok ellenőrzése
☐ kekszalag	utf8mb4_general_ci	Jogosultságok ellenőrzése
☐ monthpython	utf8_hungarian_ci	Jogosultságok ellenőrzése

- 2) XAMPP elindítása a MYSQL szerverhez, majd az üres adatbázis létrehozása.
- 3) .env környezeti állomány átírása

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=fingatlan2
DB_USERNAME=root
DB_PASSWORD=
```

```
SESSION_DRIVER=files
SESSION_LIFETIME=120
SESSION_ENCRYPT=false
SESSION_PATH=/
SESSION_DOMAIN=null
```

4) Migrációs állományok létrehozása az alábbi parancsokkal

```
PS C:\xampp\htdocs\F2Ingatlan> php artisan make:migration create_kategoriak_table

INFO Migration [C:\xampp\htdocs\F2Ingatlan\database\Migrations\2025_02_13_154209_create_kategoriak_table.php] created successfully.

PS C:\xampp\htdocs\F2Ingatlan> php artisan make:migration create_ingatlanok_table

INFO Migration [C:\xampp\htdocs\F2Ingatlan\database\Migrations\2025_02_13_154224_create_ingatlanok_table.php] created successfully.
```

5) A migrációs állományok szerkesztése az adatbázis táblái alapján (ahogy a feladatban szerepel)

```
* Run the migrations.
*/
public function up(): void
{
    Schema::create('kategoriak', function (Blueprint $table) {
        $table->id();
        $table->string('nev')->unique();
        $table->timestamps();
    });
}
```

```

| */
public function up(): void
{
    Schema::create('ingatlanok', function (Blueprint $table) {
        $table->id();
        $table->foreignId('kategoria')->constrained('kategoriak');
        $table->text('leiras')->nullable();
        $table->date('hirdetesDatuma')->nullable();
        $table->boolean('tehermentes');
        $table->integer('ar');
        $table->string('kepUrl')->nullable();
        $table->timestamps();
    });
}

```

6) A migrációs állományok futtatása az alábbi paranccsal:

```
PS C:\xampp\htdocs\F2Ingatlan> php artisan migrate
```

INFO Preparing database.

Creating migration table 42.22ms **DONE**

INFO Running migrations.

```

0001_01_01_000000_create_users_table ..... 273.47ms DONE
0001_01_01_000001_create_cache_table ..... 42.91ms DONE
0001_01_01_000002_create_jobs_table ..... 214.06ms DONE
2025_02_13_154209_create_kategoriak_table ..... 75.04ms DONE
2025_02_13_154224_createingatlanok_table ..... 137.30ms DONE

```

7) PHP API bővítmény telepítése

```

PS C:\xampp\htdocs\F2Ingatlan> php artisan install:api
./composer.json has been updated
Running composer update laravel/sanctum
Loading composer repositories with package information
Updating dependencies
Lock file operations: 1 install, 0 updates, 0 removals
- Locking laravel/sanctum (v4.0.8)
Writing lock file
Installing dependencies from lock file (including require-dev)
Package operations: 1 install, 0 updates, 0 removals
- Installing laravel/sanctum (v4.0.8): Extracting archive
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump

```

No security vulnerability advisories found.

INFO Published API routes file.

One new database migration has been published. Would you like to run all pending database migrations? (yes/no) [yes]:

> yes

INFO Running migrations.

2025_02_13_155528_create_personal_access_tokens_table .. 234.79ms **DONE**

INFO API scaffolding installed. Please add the [Laravel\Sanctum\HasApi

8) Szerver indítása

```
C:\xampp\htdocs\F2Ingatlan>php artisan serve
forking is not supported on this platform
```

INFO Server running on [http://127.0.0.1:8000].

Press Ctrl+C to stop the server

9) Adatok tömeges feltöltése Seeder-ek segítségével

```
PS C:\xampp\htdocs\F2Ingatlan> php artisan make:seeder IngatlanSeeder
```

INFO Seeder [C:\xampp\htdocs\F2Ingatlan\database\seeders\IngatlanSeeder.php] created successfully.

```
DB::table('kategoriak')->insert([
    ['nev' => 'Ház'],
    ['nev' => 'Lakás'],
    ['nev' => 'Építési telek'],
    ['nev' => 'Garázs'],
    ['nev' => 'Mezőgazdasági terület'],
    ['nev' => 'Ipari ingatlan']
]);
```

```
DB::table('ingatlanok')->insert([
    [
        'kategoria' => 1,
        'leiras' => 'Családi ház omladozó',
        'hirdetesDatuma' => '2025-01-15',
        'tehermentes' => true,
        'ar' => 4500000,
        'kepUrl' => 'https://ingatlanok.hu/65874.jpg'
    ],
    [
        'kategoria' => 2,
        'leiras' => '4. emeleti lakás',
        'hirdetesDatuma' => '2025-01-20',
        'tehermentes' => false,
        'ar' => 18000000,
        'kepUrl' => 'https://ingatlanok.hu/65875.jpg'
    ]
]);
```

A fenti Seeder-ekben a `DB::table` elé szükséges a `\` karakter!

```
PS C:\xampp\htdocs\F2Ingatlan> php artisan db:seed --class=IngatlanSeeder
```

INFO Seeding database.

10) Model létrehozása a táblák alapján

```
PS C:\xampp\htdocs\F2I
php artisan make:model
```

INFO Model [C:\xampp\htdocs\F2Ingatlan] created successfully.

```
PS C:\xampp\htdocs\F2Ingatlan> php artisan make:model Ingatlan
```

INFO Model [C:\xampp\htdocs\F2Ingatlan\app\Models\Ingatlan.php] created successfully.

```
PS C:\xampp\htdocs\F2Ingatlan> 
```

```
class Kategoria extends Model
{
    protected $table='kategoriak';
    protected $fillable = ['nev'];

    public function ingatlanok()
    {
        return $this->hasMany(Ingatlan::class, 'kategoria');
    }
}

class Ingatlan extends Model
{
    protected $table = 'ingatlanok';
    protected $fillable = ['kategoria', 'leiras', 'hirdetesDatuma', 'tehermentes',
        'ar', 'kepUrl'];

    public function kategoria()
    {
        return $this->belongsTo(Kategoria::class, 'kategoria');
    }
}
```

11) Controller létrehozása a tényleges művelethez.

```
php artisan make:controller IngatlanController
```

12) Model módosítása a feladatnak megfelelően

```
class Ingatlan extends Model
{
    use HasFactory;

    protected $table = 'ingatlanok'; // A tábla neve

    protected $fillable = [
        'kategoria', 'leiras', 'hirdetesDatuma',
        'tehermentes', 'ar', 'kepUrl'
    ];

    public $timestamps = false;

    public function kategoria(): BelongsTo
    {
        return $this->belongsTo(Kategoria::class, 'kategoria');
    }

    public static function getAllWithCategory()
    {
        return self::with('kategoria:id,nev')->get()->map(function ($ingatlan) {
            return [
                'id' => $ingatlan->id,
                'kategoria' => $ingatlan->kategoria->nev ?? null,
                'leiras' => $ingatlan->leiras,
                'hirdetesDatuma' => $ingatlan->hirdetesDatuma,
                'tehermentes' => $ingatlan->tehermentes,
                'ar' => $ingatlan->ar,
                'kepUrl' => $ingatlan->kepUrl,
            ];
        });
    }
}
```

13) Controller kezelése

```
namespace App\Http\Controllers;

use App\Models\Ingatlan;
use Illuminate\Http\Request;
use Illuminate\Http\JsonResponse;

class IngatlanController extends Controller
{
    public function index(): JsonResponse
    {
        return response()->json(Ingatlan::getAllWithCategory(), 200);
    }
}
```

14) Route API módosítása

```
use App\Http\Controllers\IngatlanController;  
  
Route::get('/ingatlan', [IngatlanController::class, 'index']);
```