Vizsgázó neve:………………………..

# Szakmai vizsga

**Projektfeladat**

Az ágazat megnevezése: **Informatika és távközlés**

Szakma megnevezése: **Szoftverfejlesztő és -tesztelő technikus szakma**

A vizsgatevékenység megnevezése:

**Asztali- és webes szoftverfejlesztés, adatbázis-kezelés feladatsor**

Időtartam: **240 perc**

A vizsgatevékenység aránya a teljes szakmai vizsgán belül:

**90%**

Jóváhagyta:

Dátum: 2022. ……………………………

……………………………..

Módos Gábor

igazgató

**2022**

Vizsgaszervező neve:

**Győri SZC Jedlik Ányos Gépipari és Informatikai Technikum és Kollégium**

Vizsga dátuma:

**--------------**

Vizsga helyszíne:

**Győri SZC Jedlik Ányos Gépipari és Informatikai Technikum és Kollégium**

**9021 Győr, Szent István út 7.**

Használható segédeszköz:

A gyakorlati vizsgatevékenység során a jelölt, a feladat kidolgozása közben az alábbi eszközöket használhatja:

* **internethasználattal rendelkező asztali számítógép**

## Grafikus és konzolos részt egyaránt tartalmazó asztali alkalmazás fejlesztése

A következő feladatban ingatlan hirdetésekkel kapcsolatos szöveges állomány áll a rendelkezésünkre, melyekkel programozási feladatokat kell megoldania.

A feladat megoldása során vegye figyelembe a következőket:

* *A képernyőre írást igénylő részfeladatok eredményének megjelenítése előtt írja a képernyőre a feladat sorszámát (például:* 3. feladat:*)!*
* *Az egyes feladatokban a kiírásokat a minta szerint készítse el!*
* *Az ékezetmentes azonosítók és kiírások is elfogadottak.*
* *Az azonosítókat kis- és nagybetűkkel is kezdheti.*
* *A program megírásakor az állományban lévő adatok helyes szerkezetét nem kell ellenőriznie, feltételezheti, hogy a rendelkezésre álló adatok a leírtaknak megfelelnek.*
* *A megoldását úgy készítse el, hogy az azonos szerkezetű, de tetszőleges bemeneti adatok mellett is helyes eredményt adjon!*

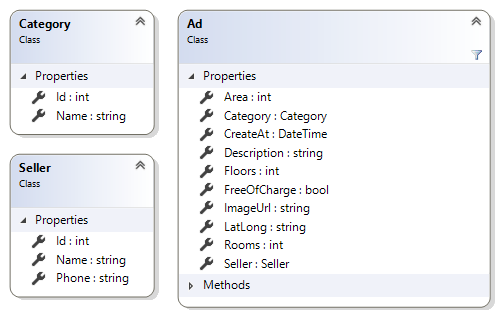
Az adatforrás két formában áll rendelkezésére, amelyek közül egyet kell választania a feladat megoldásához.

1. realestates.json elnevezésű JSON típusú szöveges fájl
2. realestates.csv elnevezésű szöveges állomány   
   Az állomány első sora a mezőneveket tartalmazza, az adatokat pontosvesszővel választottuk el.

Mindkét esetben a kapott adatok kódolása UTF-8. Az adatforrásokban a következő adatokat találja meg. Néhány esetben az adatforrás jellege miatt az elnevezések nem pontosan fedik az alábbi lista elemeit, ilyenkor értelemszerűen azonosítsa az adatokat:

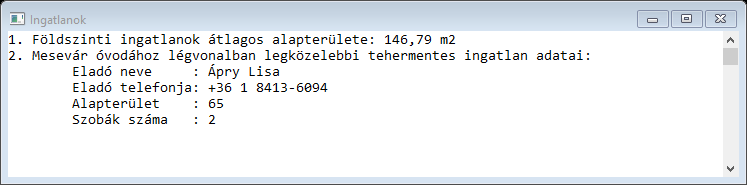
* *id: a hirdetés azonosítója, egész szám*
* *description: a hirdetés szövege, szöveges*
* *rooms: az ingatlan szobáinak száma, egész szám*
* *area: az ingatlan alapterülete, egész szám*
* *floors: az ingatlan hányadik emeleten helyezkedik el, egész szám*
* *category: az ingatlan kategóriája, önálló osztály (id, name)*
* *seller: az ingatlan eladója, önálló osztály (id, name, phone)*
* *freeofcharge: az ingatlan tehermentes-e, logikai*
* *imageUrl: a hirdetéshez tartozó képállomány elérési útvonala, szöveges*
* *latlong: az ingatlan földrajzi koordinátái (szélességi, hosszúsági fok) vesszővel elválasztva. A szélességi és hosszúsági fok valós szám, a tizedesjel a pont karakter.*
* *createAt: a hirdetés felkerülésének dátuma, dátum*

1. Készítsen konzolos alkalmazást a következő feladatok megoldására, melynek projektjét RealEstate néven mentse el!
2. Készítsen saját osztályokat Ad, Seller és Category azonosítóval, melynek az adattagjainak azonosítóit és láthatósági szintjét az osztálydiagramok szemléltetik! A privát adattagokat egy lakat szimbólum különbözteti meg a publikusaktól. Ékezetes azonosítókat is készíthet, illetve azokat kis- és nagybetűkkel is kezdheti.



1. Készítsen statikus metódust az Ad osztályhoz az adatforráshoz történő kapcsolódásra, és az adatforrásban levő adatok betöltésére! A metódus neve utaljon az adatforrásból betöltésre (pl.: LoadFromCsv). A metódus visszatérési értéke az Ad osztályból képzett lista legyen! A metódus paramétere a betöltendő fájl neve legyen!
2. Szükség esetén az Ad osztály adattagjainak beállítását konstruktor segítségével állítsa be! A paraméterben kapott adatok az adatforrásoktól függően változóak lehetnek.
3. Olvassa be a választott adatforrás adatait és tárolja az adatokat az Ad osztály segítségével egy olyan összetett adatszerkezetben, amely használatával a további feladatok megoldhatók!
4. Határozza meg és írja ki a minta szerint az eladásra kínált földszinti ingatlanok átlagos alapterületét! Az eredményt a minta alapján két tizedesjegy pontossággal írja ki!
5. Készítsen DistanceTo néven valós értékkel visszatérő metódust, amelynek segítségével adott ingatlan esetében meg lehet állapítani egy GPS koordinátától való távolságot Pitagorasz-tétel segítségével!
6. A Mesevár óvoda Budán a 47.4164220114023, 19.066342425796986 GPS koordinátán helyezkedik el. Keresse ki és írja ki a minta alapján annak a tehermentes ingatlannak az adatait, melyik a legközelebb van légvonalban a Mesevár óvodához!

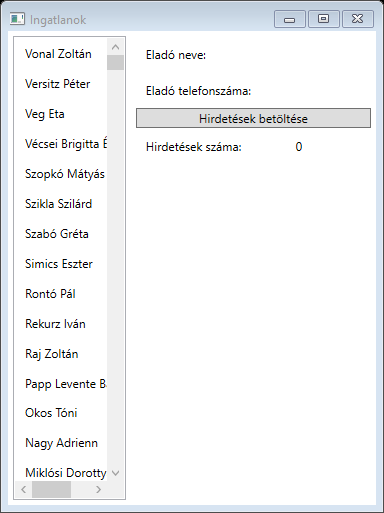
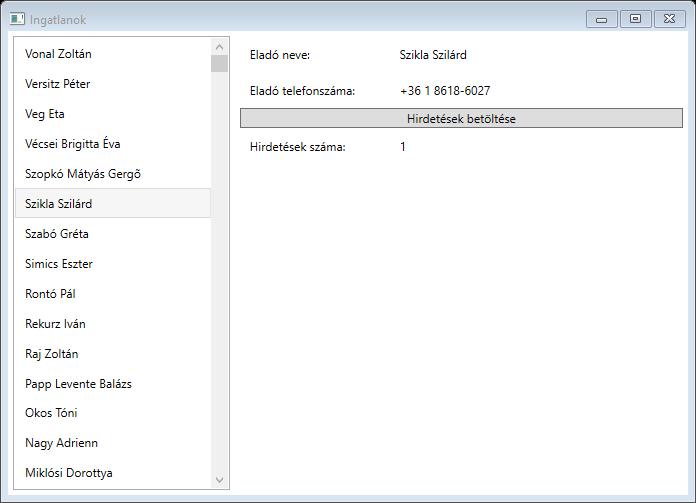
## Minta

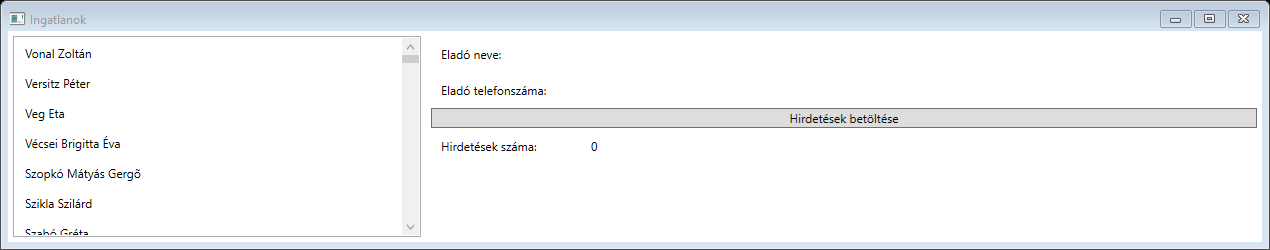


A grafikus alkalmazás elkészítése előtt hozzon létre a lokális adatbáziskezelő rendszerében ingatlan néven UTF-8 kódolású, magyar egybevetésű adatbázist! Ezt követően futassa le az ingatlan.sql szkriptet az ingatlan adatbázisban!

1. Készítsen grafikus alkalmazást a következő feladatok megoldására, melynek projektjét RealEstateGUI néven mentse el!
2. A megjelenő ablak a minták alapján reszponzív viselkedésű legyen! Az ablakban levő vezérlők típusai és a feliratok feleljenek meg a mintának!
3. A Seller és Ad osztályokat a RealEstate projekt segítségével hivatkozza be!
4. Csatlakozzon az ingatlan adatbázishoz! A mintán szereplő bal oldali listába töltse be a sellers tábla összes rekordját! A listában csak a vevők nevei jelenjenek meg!
5. Az „Eladó neve” címke mellett a lista kijelölt eleme jelenjen meg adatkötés segítségével!
6. Az „Eladó telefonszáma” címke mellett a lista kijelölt elemének telefonszáma jelenjen meg adatkötés segítségével!
7. A „Hirdetések betöltése” gombra kattintva az „Hirdetések száma” címke mellett a lista kijelölt eleme alapján jelenjen meg, hogy az adott személyhez hány hirdetés tartozik!

## Minták



## Reszponzív viselkedésű weboldal

A következő feladatban egy ingatlanok értékesítésével foglalkozó vállalkozás weboldalát kell módosítania a feladatleírás és a minta szerint. Ahol a feladat másként nem kéri, a formázási beállításokat a style.css stílusállományban végezze el úgy, hogy az új szelektorokat az állomány végén helyezze el!

Nagyobb felbontású, színes mintákat a kész weboldalról a minta01.png, minta02.png és a minta03.png állományokban talál, melyet tilos a megoldásában felhasználni!

Nyissa meg az index.html, newad.html, offers.html és a style.css állományokat és szerkessze azok tartalmát az alábbiak szerint:

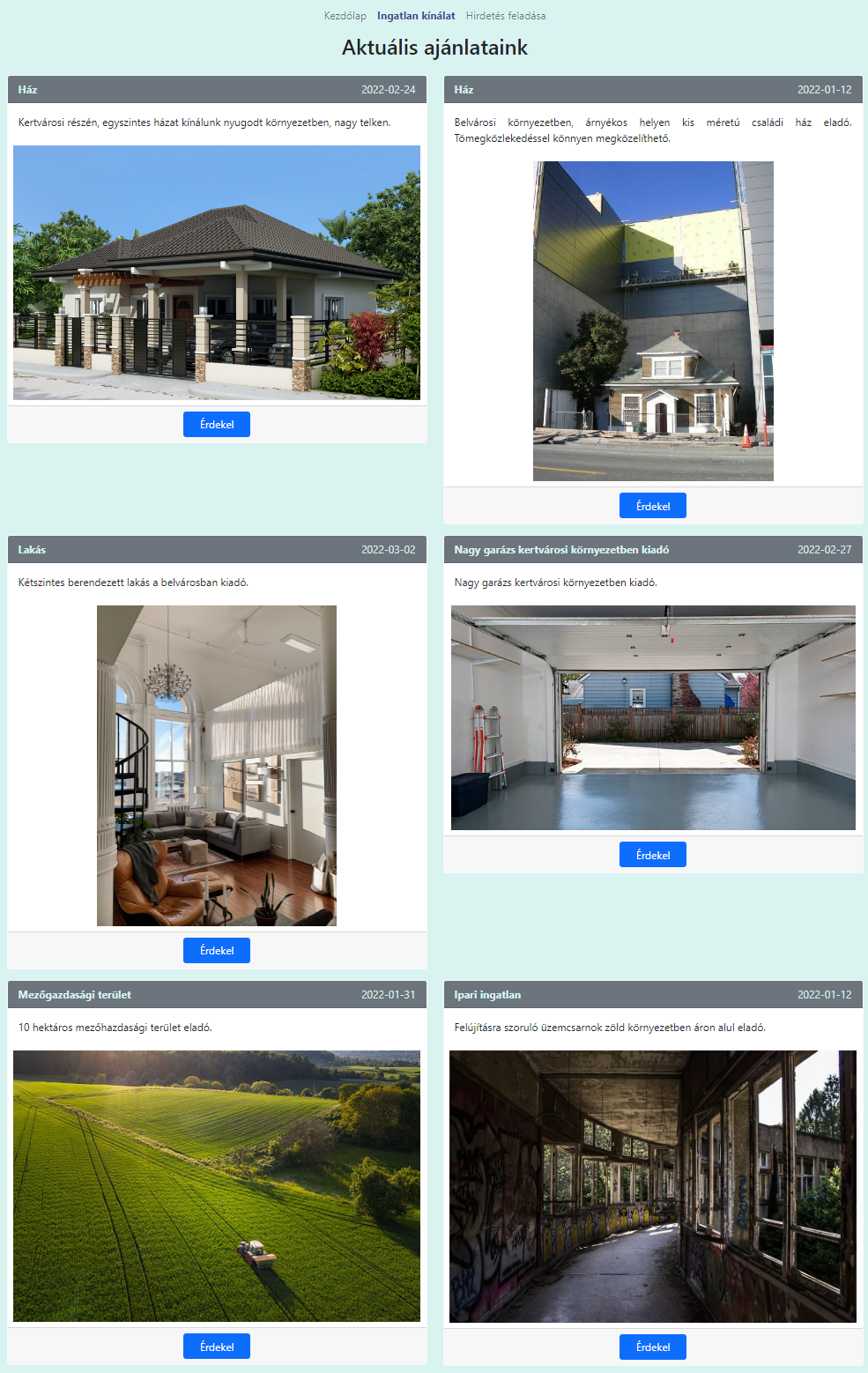
* A style.css állományban megfelelő szelektor alkalmazásával állítsa be, hogy az ingatlan kínálat oldalon a kártyák címe (pl.: Ház 2022-02-24) feliratnak a szövegszíne ugyanolyan legyen, mint az oldal háttérszíne!
* Mindhárom oldalon a navigációs sávon kiemelten látszik az aktuális oldalhoz tartozó menüpont (az aktuális sötétebb szürke, mint a másik kettő). A stíluslapon a megfelelő szelektor segítségével állítsa be, hogy az aktív laphoz tartozó menüpont színe (R: 71, G: 71, B:150) legyen, és félkövéren jelenjen meg!
* Állítsa be a stíluslapon, hogy a start osztálykijelölőhöz tartozó keret háttérképe ne ismétlődjön, és teljesen töltse ki a keretet! (Az ismétlődés nagyobb képernyőfelbontás esetén jelentkezik.)
* A nyitó oldalon (index.html) található gombokhoz tartozó HTML kódot módosítsa úgy, hogy rájuk kattintva azonos lapon megnyíljon az offers.html illetve a newad.html oldal!
* A newad.html oldalon az oldal tetején a menü nem középen jelenik meg. Javítsa a hibát!
* A newad.html oldal végén a két soros javascript kód feladata, hogy a hirdetés dátumának értéke az aktuális dátum legyen. A script futásakor a konzolon az alábbi hiba jelenik meg: Uncaught TypeError: document.querySelector(...) is null

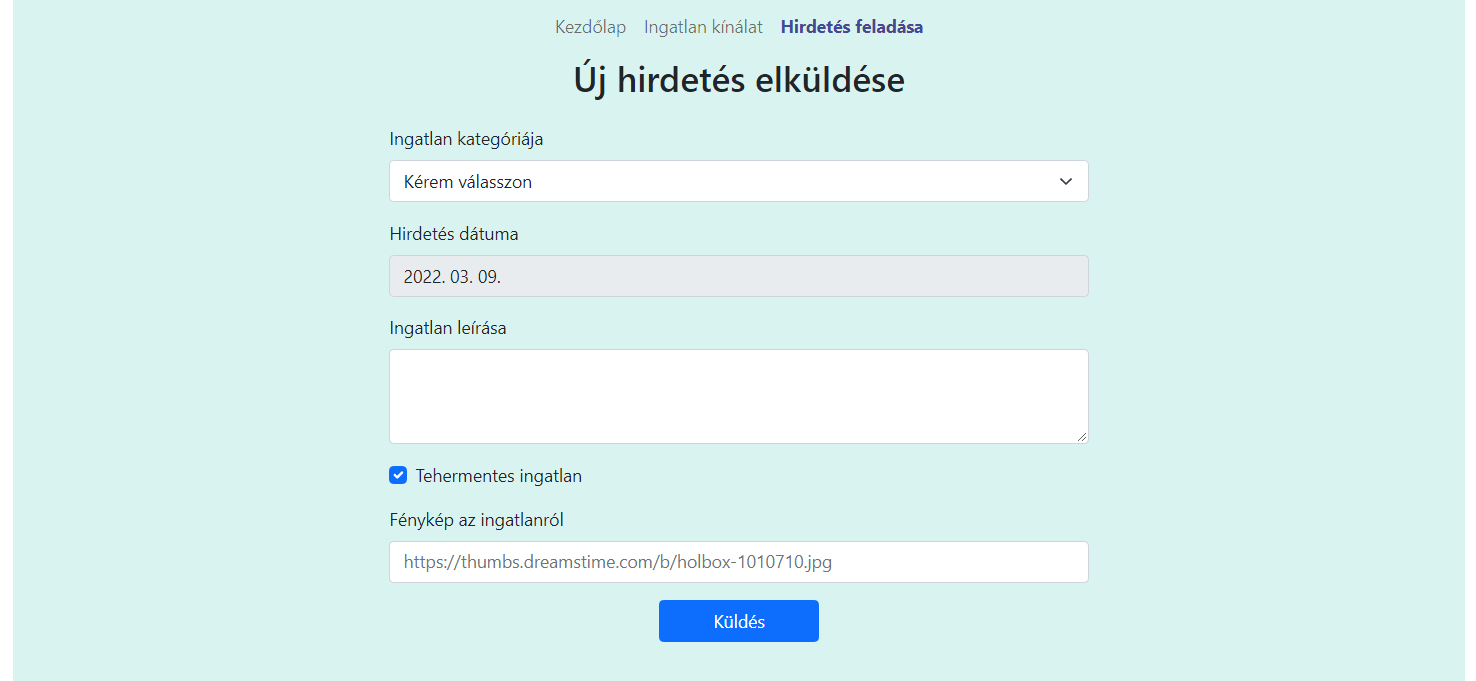
Javítsa a hibát!

* A kínálatot tartalmazó oldalon szúrjon be a két ház után egy újabb hirdetést! A hirdetés adatait megtalálja a lakas.txt állományban. Az állományban szereplő minden adatra szüksége van a teljeskörű megoldáshoz.
* Egészítse ki az új hirdetés feladása oldalt egy új beviteli mezővel:
  + A beviteli mező címkéjének a felirata legyen: „Fénykép az ingatlanról”!
  + A beviteli mező típusa legyen URL!
  + Ügyeljen arra, hogy a megjelenő elemek stílusa megegyezzen az oldalon lévő többi elem stílusával!

# Minták



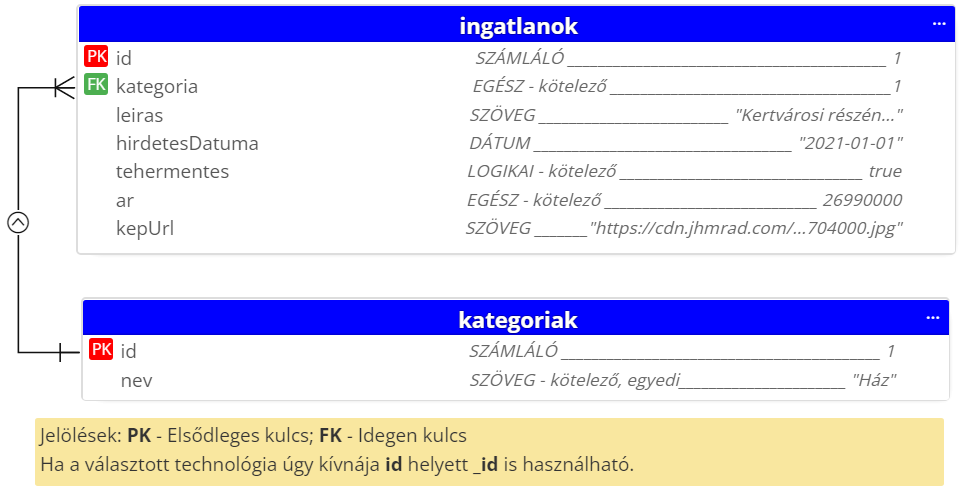




## Backend programozás

Az alábbi feladatban egy ingatlan hirdetéssel foglalkozó cég weboldalának backend szerverét kell elkészítenie.

1. Hozzon létre backend szerver projektet az Ön által választott programnyelven, illetve fejlesztési környezetben! A projektmappát „Vezetéknév\_Keresztnév\_backend” formában nevezze el! A feladat megoldása során ékezetmentes azonosítókat és állományneveket is használhat.
2. Készítsen adatbázist ingatlan néven!
3. A lenti ábra alapján:
   1. Hozzon létre modelleket a megfelelő adattípusokkal és beállításokkal!
   2. Az ingatlan adatbázisban hozza létre a két táblát!



1. Oldja meg, hogy ha a hirdetés dátuma nem kerül megadásra, akkor annak alapértelmezett értéke az aktuális dátum legyen!
2. Töltse fel a kategoriak táblát (kollekciót) az alábbi rekordokkal (dokumentumokkal):

|  |  |
| --- | --- |
| **id** | **nev** |
| 1 | Ház |
| 2 | Lakás |
| 3 | Építési telek |
| 4 | Garázs |
| 5 | Mezőgazdasági terület |
| 6 | Ipari ingatlan |

*A létrehozott végpontok teszteléséhez – a választott technológiától függően – az* ingatlanok *táblához tartozó* ingatlanok.csv *ill.* ingatlanok.json *fájlban található minta adatokat használhatja.*

*A további feladatok megoldása során ügyeljen arra, hogy a megoldása az elvárt végpontokon kívül egyéb végpontokat ne tartalmazzon! A végpontok funkcionalitását megvalósító kontroller(ek)ben a szükséges metódusokon kívül egyéb (felesleges) kódokat ne hozzon létre!*

1. Hozzon létre végpontot az adatbázisban szerepelő összes ingatlan lekérdezésére az alábbi beállításokkal! Oldja meg, hogy a válasz üzenetben a kategoriak tábla nev mezője is szerepeljen! *A válasz tartalma a lenti mintáktól eltérhet a választott technológiának megfelelően, de a mintákban szereplő összes adatot – beleértve a kategória nevét is – tartalmaznia kell!*

|  |  |  |  |
| --- | --- | --- | --- |
| Metódus | URL | Body | Válasz |
| GET | /api/ingatlan | üres | JSON |

Lehetséges válasz üzenet (**200 OK** státuszkóddal):

[

{

"id": 1,

"kategoria": "Lakás",

"leiras": "Eladó ...",

"hirdetesDatuma": "2022-03-10",

"tehermentes": true,

"ar" : 26990000

"kepUrl": "https://pictures.com/apartman.jpg"

}

...

]

vagy:

[

{

"id": 1,

"kategoria": {

"id": 2,

"nev": "Lakás"

},

"leiras": "Eladó ...",

"hirdetesDatuma": "2022-03-10",

"tehermentes": true,

"ar" : 26990000

"kepUrl": "https://pictures.com/apartman.jpg"

}

...

]

1. Hozzon létre végpontot új ingatlan rögzítésére az alábbi beállításokkal! Sikeres rögzítés esetén **201 CREATED** státuszkóddal és az újonnan beszúrt rekord azonosítóját tartalmazó JSON üzenettel térjen vissza! Bármely kötelező mező hiánya esetén **400 BAD REQUEST** hibakóddal és a hibára utaló „Hiányos adatok” üzenettel térjen vissza!

|  |  |  |  |
| --- | --- | --- | --- |
| Metódus | URL | Body | Válasz |
| POST | /api/ingatlan | JSON | JSON/szöveges |

Példa a kérés törzsében elküldött JSON tartalmára:

{

"kategoria": 2,

"leiras": "Eladó ...",

"hirdetesDatuma": "2022-03-19",

"tehermentes": true,

"ar": 45000000

"kepUrl": "https://pictures.com/apartman.jpg"

}

vagy:

{

"\_id": 99,

"kategoria": 2,

"leiras": "Eladó ...",

"hirdetesDatuma": "2022-03-19",

"tehermentes": true,

"ar": 45000000

"kepUrl": "https://pictures.com/apartman.jpg"

}

Válasz üzenet (**201 CREATED** státuszkóddal):

{

"Id": 7

}

Egy lehetséges válasz üzenet (**400 BAD REQUEST** státuszkóddal):

"Hiányos adatok."

1. Hozzon létre végpontot az adatbázisban szerepelő ingatlan törlésére az alábbi beállításokkal! Az {id} helyén a törölni kívánt ingatlan azonosítója szerepeljen! Sikeres törlés esetén **204 NO CONTENT** státuszkóddal térjen vissza! Amennyiben a megadott azonosító nem létezik **404 NOT FOUND** státuszkóddal és az „Ingatlan nem létezik” üzenettel térjen vissza!

|  |  |  |  |
| --- | --- | --- | --- |
| Metódus | URL | Body | Válasz |
| DELETE | /api/ingatlan/{id} | üres | üres/szöveges |

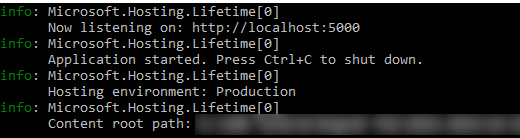
Egy lehetséges válasz üzenet (**404 NOT FOUND** státuszkóddal):

"Az ingatlan nem létezik."

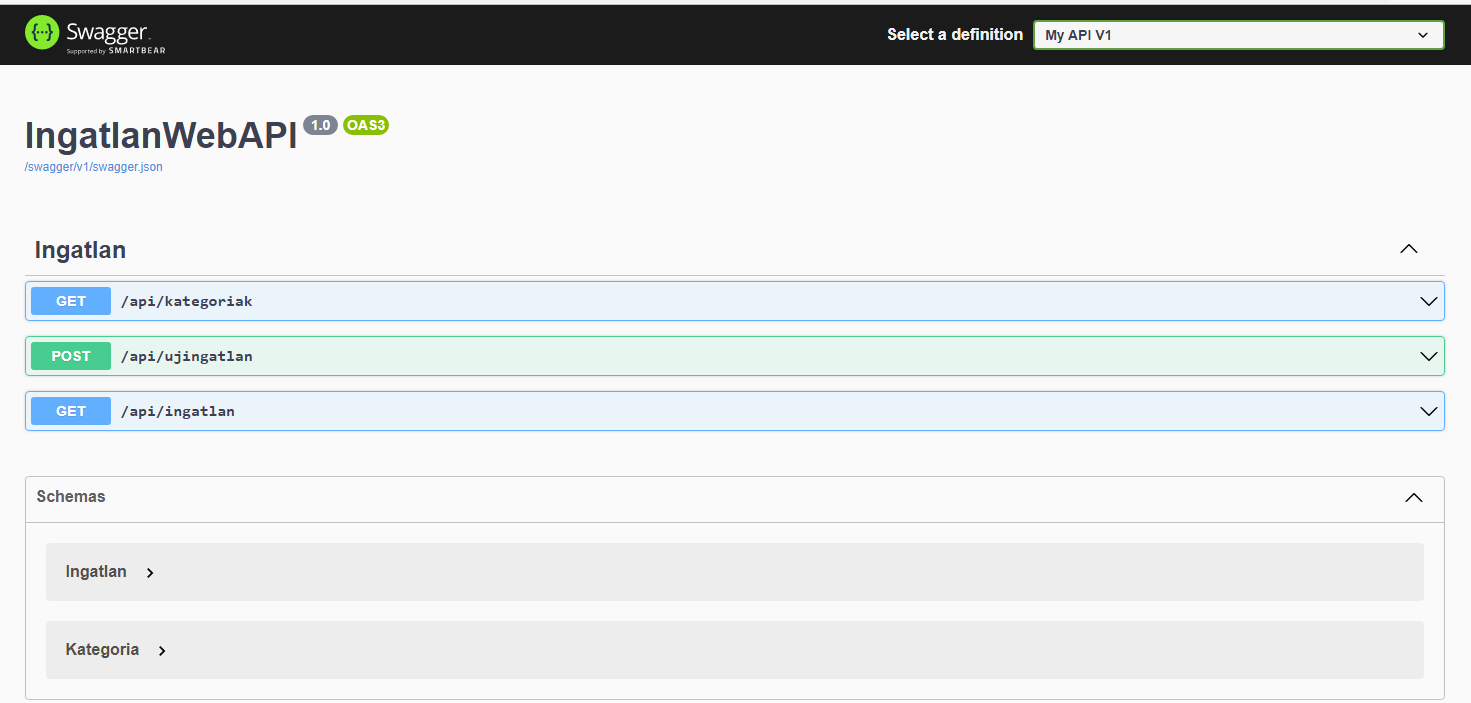
1. Hozzon létre Thunder Client vagy Postman kollekciót Backend teszt néven, melyben az Ön által létrehozott összes útvonalat ellenőrzi! Exportálja a kollekciót a projekt gyökér könyvtárába Backend\_teszt néven!
2. Projektmappáját a választott technológiának megfelelően adja le:
   1. Tömörítés előtt törölje a felesleges állományokat! **Ügyeljen arra, hogy feladatmegoldást tartalmazó mappát/állományt ne töröljön!**
   2. A forráskódját tömörítse be Vezetéknév\_Keresztnév\_backend.zip néven!
   3. A tömörített állomány tartalmazza az adatbázist létrehozó SQL scriptet vagy JSON állományt is!

## Frontend programozás

A következő feladatban egy frontend alkalmazást készítését kell elvégeznie a kiadott leírás szerint! Az alkalmazás felhasználói felületének kialakítását minták segítik, melyeket **nem kell** követnie a megoldásában. Indítsa el a projektkönyvtárának backend mappájából a weboldalát kiszolgáló szervert (IngatlanWebAPI.exe)! A program indulása után az alábbi képernyő jelenik meg:



A mintán látható „Content root path” a program indításának a helyét mutatja. A webkiszolgáló a http://localhost:5000 címen fog elindulni. A böngészőbe beírva a <http://localhost:5000/docs> URL-t a kap segítséget a webszerveren elérhető REST API függvényekhez.



Az alábbi REST API függvények érhetőek el a szerveren:

* /api/kategoriak

Metódus: GET

Válasz: Az adatbázisban lévő kategóriák azonosítói és nevei.

Válasz minta:

[

{

"id": 1,

"megnevezes": "Ház"

},

...

{

"id": 6,

"megnevezes": "Ipari ingatlan"

}

]

* /api/ingatlan

Metódus: GET

Válasz: Az adatbázisban jelenleg található hirdetések adatai.

Válasz minta:

[

{

"id": 1,

"kategoriaId": 1,

"kategoriaNev": "Ház",

"leiras": "Kertvárosi részén, egyszintes házat kínálunk ...",

"hirdetesDatuma": "2022.02.27",

"tehermentes": true,

"kepUrl": "https://cdn.jhmrad.com/wp-content/uploads...jpg"

},

...

{

"id": 6,

"kategoriaId": 6,

"kategoriaNev": "Ipari ingatlan",

"leiras": "Felújításra szoruló üzemcsarnok zöld környezetben ...",

"hirdetesDatuma": "2022.03.01",

"tehermentes": false,

"kepUrl": "https://cdn.pixabay.com/photo/2019/01/31/09/...jpg"

}

]

* /api/ujingatlan

Metódus: POST

A kérés törzsében kell a szerver felé elküldeni a rögzíteni kívánt új hirdetés adatait.

Minta a kérésre:

{

"kategoriaId": 1,

"leiras": "Szép ház van eladó",

"hirdetesDatuma": "2022-03-09",

"tehermentes": true,

"kepUrl": "http://kep.hu/azenhazam.png"

}

Válasz minta:

{

"id": 7,

"kategoriaId": 1,

"kategoriaNev": "Ház",

"leiras": "Szép ház van eladó",

"hirdetesDatuma": "2022.03.09",

"tehermentes": true,

"kepUrl": "http://kep.hu/azenhazam.png"

}

**FONTOS! A szerverhez nem tartozik fizikai adatbázis, a program újraindításakor az adatok „visszaállnak” az eredeti állapotukba.**

## Feladatok

Az Ön által választott Javascript keretrendszer (Angular, VueJs vagy React) valamelyikével készítse el az alábbi feladatokat:

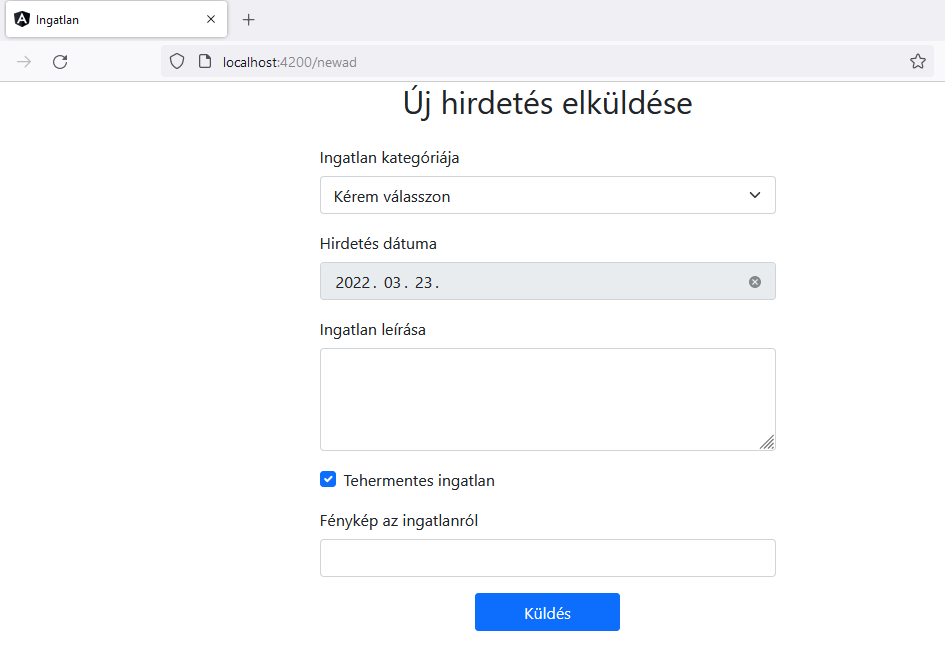
1. Adja hozzá az alkalmazáshoz a Bootstrap keretrendszer 5-ös verzióját!
2. Hozzon létre egy komponenst, amely a webalkalmazás nyitó oldala lesz:
   1. A komponenst a minta alapján hozza létre!
   2. A komponens létrehozása során használhatja a források könyvtárban szereplő openpage.html oldalt és openpage.css stílusállományt!
   3. Ügyeljen arra, hogy az oldal háttérképe a – szintén a források könyvtárban található – real-estate-agent.png kép legyen!
3. Hozzon létre egy újabb komponenst a hirdetések megjelenítéséhez!
   1. Az oldalon szereplő ajánlatokat tartalmazó adatokat a kiszolgáló szerver /api/ingatlan URL-jén szereplő REST API függvénnyel kérdezze le!
   2. Az adatokat táblázatos formában – a kiadott mintára hasonlító – az Ön által választott keretrendszerhez alkalmazható külső komponens segítségével jelenítse meg! Ügyeljen arra, hogy a táblázatban szereplő adatok és az oszlopok sorrendje egyezzen a mintával!
   3. Állítsa be a komponensnek az eléréséhez a /offers útvonalat!
   4. A kezdőoldalon a „Nézze meg kínálatunkat” feliratú gombhoz állítsa be, hogy arra kattintva navigáljon át a /offers útvonalra!
4. Hozza létre az új hirdetés feladását megvalósító komponenst!
   1. A komponens létrehozása során használhatja a források könyvtárban szereplő newad.html oldalt.
   2. Az ingatlanok típusait tartalmazó lenyíló listához az elemeket a szerver /api/kategoriak URL-jéről töltse le!
   3. Állítsa be a hirdetés dátuma mező értékét csak olvashatóra, a mező értéke az oldal megnyitásakor legyen az aktuális dátum!
   4. Az új ajánlatokat küldje el a szerver /api/ujingatlan URL-re az API leírás alapján!
   5. Sikeres küldés esetén navigáljon át a /offers útvonalra, hiba esetén a mintának megfelelően jelenítse meg az üzenetet!
   6. Állítsa be ennek a komponensnek az eléréséhez a /newad útvonalat!
   7. A kezdőoldalon a „Hirdessen nálunk!” feliratú gombhoz állítsa be, hogy arra kattintva navigáljon át a /newad útvonalra!
5. Forráskódját a választott technológiának megfelelően adja le:
   1. Tömörítés előtt törölje a felesleges állományokat! **Ügyeljen arra, hogy feladatmegoldást tartalmazó mappát/állományt ne töröljön!**
   2. A forráskódját tömörítse be Vezetéknév\_Keresztnév\_frontend.zip néven!

## Minták



A képen szöveg látható

Automatikusan generált leírás



A képen szöveg látható

Automatikusan generált leírás