

JEGYZŐKÖNYV

Számítógép Architektúrák

Féléves feladat
2023 ősz féléves feladat

Készítette: Tóth Zsombor Gábor

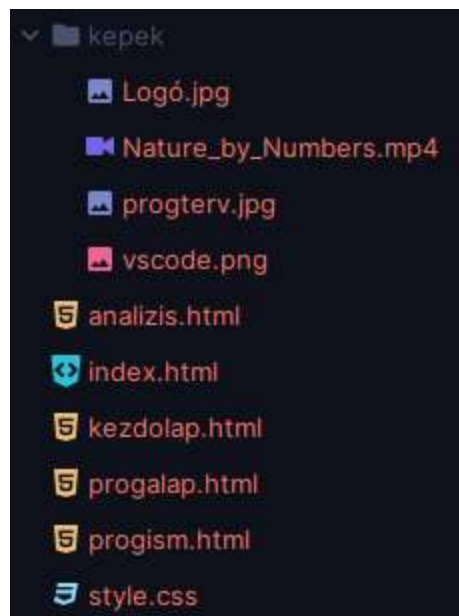
Neptunkód: D0H157

Dátum: 2023.12.10

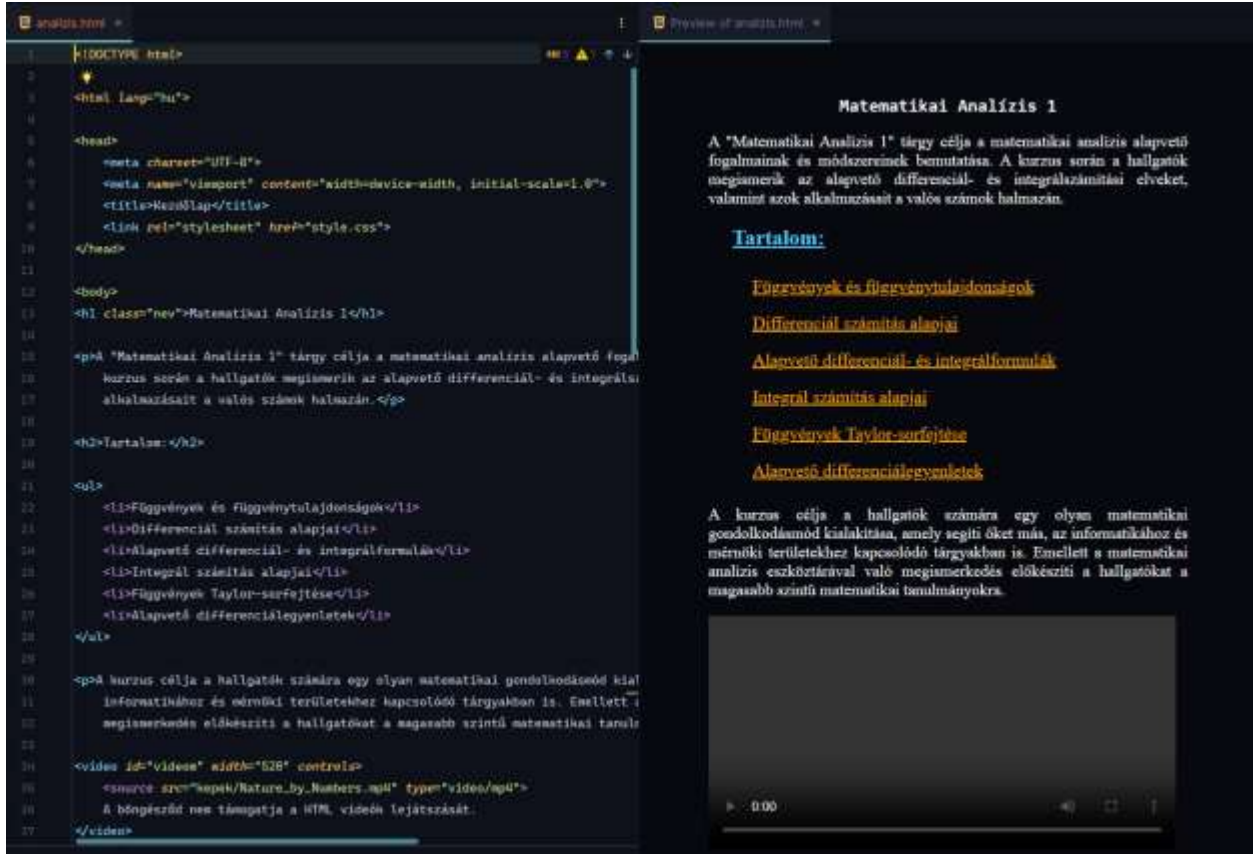
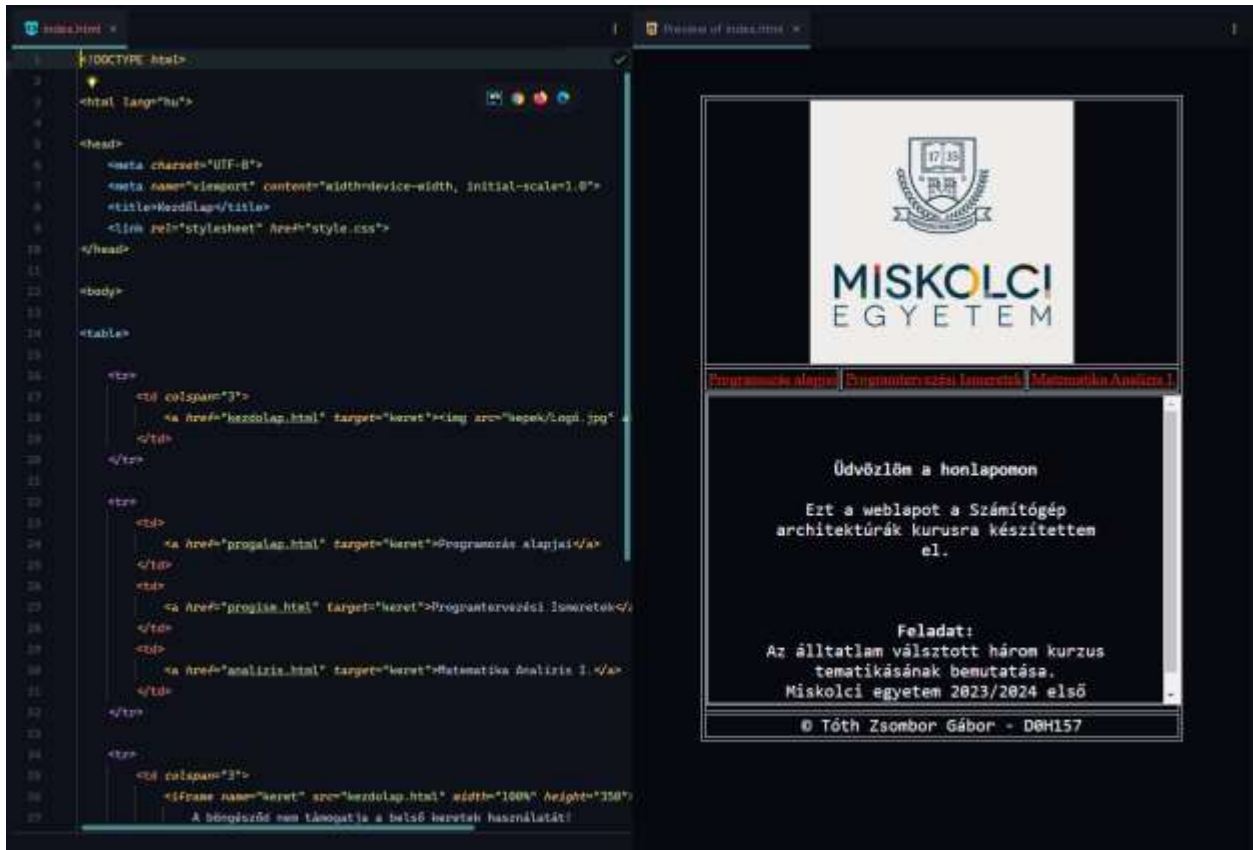
1. Feladat

Tervezzen meg egy HTML weblapot, használjon modern webfejlesztési elemeket (HTML4/HTML5, CSS, JavaScript). Készítsen egy statikus HTML4/HTML5 oldalt, amely az Ön által tanult/választott három tantárgy tematikáját mutatja be, majd illesszen be egy video fájlt egy oldalra. Legyen egy kezdőlap, amely a három kurzus menüpontját tartalmazza, ill. a fejléc képet is tartalmazzon. A kezdőlapon egy-egy kurzus menüpontjára kattintunk, megjelenik az adott kurzus tematikájának leírása és hozzá egy kép. Legyen lehetőség visszalépni a kezdőlapra. A lábléc tartalmazzon a weblap készítőjének nevét és neptunkódját. Javasolom: a forrásfájl neve megegyezik az oldal nevével (pl.: kezdolap.html, index.html etc...) Legyen egy mappa a képeknek.

A weblap struktúrája az alábbi módon néz ki:



Ezen kívül az index.html fájlom az alábbi módon néz ki, valamint a mellette lévő többi .html fájl a menürendszer alatti iframe szekcióba töltjük be amint a linkre kattintunk. Az analisis.html -t megmutatja láthatjuk, hogy nem kellett az egész oldalt újrainni, csak annyit amennyit be szeretnénk illeszteni az iframe ablakunkba. Egy utolsó képet is beillesztettem, ami megmutatja a kettőnek a kombinációját.





MISKOLCI
EGYETEM

[Fizika](#)

[Fizika](#)

[Fizika](#)

Matematikai Analízis 1

A "Matematikai Analízis 1" tárgy célja a matematikai analízis alapvető fogalmainak és módszereinek bemutatása. A kurzus során a hallgatók megismerik az alapvető differenciál- és integrálszámítási elveket, valamint azok alkalmazását a valószínűségek leírására.

Tartalom:

[Első rész: Differenciálszámítás](#)

[Differenciálható függvények](#)

[Alapvető differenciál- és integrálszámítás](#)

[Integrálható függvények](#)

2.Feladat

Az MVK Zrt. elérhetővé tesz egy szabványos GTFS adatbázist a fejlesztők számára, hogy a menetrendi adatokat a saját applikációba tudják integrálni. Írjon egy shell script fájlt, amely letölti ezt az adatbázist, és kilistázza belőle azokat az utakat, amelyek a Lakóhelyből indulnak, ill. a Lakóhelybe érkeznek! Parancsok: wget, unzip (kitömörítésre), cat, grep Szükséges ellenőrzések! Ha a letöltendő fájl már egyszer le volt töltve, akkor az újbóli letöltés előtt törölje az előző verziót! 10 Ha egy mappába már korábban ki lett tömörítve a letöltött állomány, akkor az újbóli kitömörítés törölje a mappa tartalmát! Ha nincs ilyen menetrendi adatok, akkor Miskolc várost használják a feladat elkészítéséhez!

A szkriptem több részből áll;

```
3 FILE="gtfs.zip"
4 URL="https://gtfsapi.mvkzrt.hu/gtfs/gtfs.zip"
5 DIR="${FILE%.*}"
```

Változókként megadjuk pár dolgot, legfőbbképp olyanokat amik később többször is elő fognak fordulni, esetleg változtatni szeretnénk az adott argumentumokon.

```
# Funkció a GTFS fájl letöltéséhez és kibontásához
download_and_extract_gtfs() {
    echo "Letöltjük a $FILE fájlt a $URL címről ..."
    wget "$URL" && unzip -o "$FILE" -d "$DIR"
}
```

Wget paranccsal megpróbáljuk letölteni az adott fájlt.

```
# Funkció annak ellenőrzésére, hogy létezik-e könyvtár, és törli a tartalmát
check_and_clear_directory() {
    if [ -d "$1" ]; then
        echo "A $1 mappa már létezik, töröljük a tartalmát ..."
        rm -r "$1"/*
    else
        echo "A $1 mappa még nem létezik, létrehozuk ..."
        mkdir "$1"
    fi
}
```

Egyszerű ellenőrzés a mappa létezésére.

```
# Ellenőrizze, hogy létezik-e már a GTFS fájl, és szükség esetén töltsse le/
csomagolja ki
if [ -f "$FILE" ]; then
    echo "A $FILE fájl már létezik, töröljük..."
    rm "$FILE"
fi

check_and_clear_directory "$DIR"
```

Ha a gtfs.zip fájlunk már létezik akkor törli, egyébként mindent esetben le fogja tölteni.

```
# Ellenőrizze, hogy a letöltés és a kicsomagolás sikeres volt-e
if [ $? -eq 0 ]; then
```

Megnézzük, hogy eddig a pontig minden jó-e, ha igen akkor letöltéssel, törléssel nem kell már foglalni, struktúráilag meg lett a cél.

```
echo "Adjon meg egy megállót (Búza tér, Repülőtér): "
read STOP_NAME

if grep -qF "$STOP_NAME" "$DIR/routes.txt"; then
    INFO=$(awk -F ',' -v search="$STOP_NAME" '$0 ~ search {gsub(/^[^,]*,[^,]*,/, "", $0); print $0}' "$DIR/routes.txt")
```

Bekérünk a felhasználótól egy megállót és megnézzük, hogy a routes.txt fájl-ban létezik-e.

```
IFS=$'\n' read -d '' -ra MEGALLO_ARRAY <<< "$(echo "$INFO" | cut -d ',' -f 2)"
IFS=$'\n' read -d '' -ra BUSZ_SZAM_ARRAY <<< "$(echo "$INFO" | cut -d ',' -f 1)"

BEFORE_STOP=()
AFTER_STOP=()

for ((i=0; i<${#MEGALLO_ARRAY[@]}; i++)); do
    STOP_POSITION=$(awk -F ' - ' -v search="$STOP_NAME" '{print index($0, search)}'
    <<< "${MEGALLO_ARRAY[i]}")
    if (( STOP_POSITION > 1 )); then
        BEFORE_STOP+=(" - ${MEGALLO_ARRAY[i]} - (${BUSZ_SZAM_ARRAY[i]})")
    else
        AFTER_STOP+=(" - ${MEGALLO_ARRAY[i]} - (${BUSZ_SZAM_ARRAY[i]})")
    fi
done
```

Ha létezik akkor az argumentumot amit visszkapunk felbontjuk több részre. Pl:

Repülőtér input estén visszkapjuk egyik példának, hogy:

31,15172,8,Repülőtér - Hűtőház,3

Ilyenkor a kódunk szétválogatja a visszakapott értékeket, legfőbbképp, hogy hanyas busz és a két megállót.

Busz szám: 8

Megálló: Repülőtér - Hűtőház

Ez után megnézzük, hogy az inputunk a '-' előtt vagy után van-e jelen és az alapján formázzuk meg az output-ot. Ezt minden lehetséges esetben vizsgáljuk és beleszúrjuk őket külön tömbökbe.

```
echo "Járatok amik innen indulnak:"
for entry in "${AFTER_STOP[@]}"; do
    echo "$entry"
done

echo "Járatok amik ide mennek:"
for entry in "${BEFORE_STOP[@]}"; do
    echo "$entry"
done
```

Valamit kiíratjuk őket.

Hibás esetek előfordulásánál a kódunk segít megmondani a hiba forrását.

```
else
    echo "A megadott megállóhoz nincs járat az adatbázisban."
fi
else
    echo "A letöltés vagy a kitömörítés sikertelen volt, nem tudjuk bekérni a
megállót ... "
fi
```

Mindezt egyesítve lássuk, hogy hogyan is működik:


```

A gtfs mappa még nem létezik, létrehozuk...
Letöltjük a gtfs.zip fájlt a https://gtfsapi.mvkzrt.hu/gtfs/gtfs.zip címről...
--2023-12-10 19:34:51-- https://gtfsapi.mvkzrt.hu/gtfs/gtfs.zip
Resolving gtfsapi.mvkzrt.hu (gtfsapi.mvkzrt.hu) ... 84.206.68.59
Connecting to gtfsapi.mvkzrt.hu (gtfsapi.mvkzrt.hu)[84.206.68.59]:443 ... connected.
HTTP request sent, awaiting response ... 200 OK
Length: 1056103 (1.0M) [application/zip]
Saving to: 'gtfs.zip'

gtfs.zip          100%[=====>] 1.01M 1.73MB/s   in 0.6s

2023-12-10 19:34:52 (1.73 MB/s) - 'gtfs.zip' saved [1056103/1056103]

Archive:  gtfs.zip
  inflating: gtfs/feed_info.txt
  inflating: gtfs/agency.txt
  inflating: gtfs/shapes.txt
  inflating: gtfs/routes.txt
  inflating: gtfs/stops.txt
  inflating: gtfs/trips.txt
  inflating: gtfs/stop_times.txt
  inflating: gtfs/calendar_dates.txt
Adjon meg egy megállót (Búza tér, Repülőtér):

```

Megadtam két példát, amiket jól látható, hogy pontosan mit is csináltam.

```

Adjon meg egy megállót (Búza tér, Repülőtér):
Búza tér
Járatok amik innen indulnak:
- Búza tér - Berzsenyi Dániel utca - (3A)
- Búza tér - Felsőzsolca - (7)
- Búza tér - Görömbölyi pincesor - (4)
- Búza tér - Bábonyi-bérc - (11)
- Búza tér - Szirma - (3)
- Búza tér - Takata - (43)
- Búza tér - Tampere városrész - (28)
Járatok amik ide mennek:
- AUCHAN Pesti út - Búza tér - (45)
- Déli Ipari Park - Búza tér - (53)
- Miskolctapolca - Búza tér - (20A)

```