

**Játékindítása: A testquestionsandhelp.py központi file-ból.**

## **Osztályok:**

### **Questions:**

Az említett osztály a **questionclass.py modulban** található. Azért kellett egy osztályba rakni a beolvasott adatokat, mert a **csv- fájlból** történő beolvasás után amennyiben nem objektumokba lettek volna tárolva, nagyon nehézkes lett volna az egyes összetartozó adatok keresése.

A kérdések osztályban 7db elem van ezek a következők sorrendben nehézség, kérdés, a 4db válaszlehetőség(a,b,c,d), illetve a helyes válasz. A nehézségre a listába töltéshez van szükségünk, illetve, ha a felhasználó egy adott nehézségi szinten eléri a 10. kérdést azt követően az utolsó 6 kérdés, már a nehézségi szintnek megfelelő legnehezebb kérdéseket kaphatja. A többi adat a kérdés kiírásához, illetve ellenőrzés miatt szükséges. A kiírás miatt kellett mind a két osztályban meghívnom a sztringé konvertáló speciális függvényt.

### **Resultclass:**

Az említett osztály a **questionclass.py modulban** található. Miután megszületnek egy játék végén az eredmények, az osztálynak megfelelő formátumban kerülnek tárolásra, így a későbbi feldolgozás során, például amikor kiírásra kerül osztályként könnyebben dolgozhatunk az adatokkal. Röviden az eredmények listába töltéséhez, illetve kiírásához tartalmazza a nevet a nyeresemény nagyságát és időt.

## **Függvények:**

### **menu:**

Az említett függvény a **testquestionsandhelp.py központi file-ban** található. Itt a főmenüben az alábbi elemek kerülnek megjelenítésre és a felhasználó ezek közül választhat.

„Játék indítása” opció választása esetén a **test()** függvény kerül meghívásra ezzel indul tulajdonképpen a játék.

„Dicsőségtábla” választása esetén a **resultlist()** függvény kerül meghívásra kerül meghívásra és megjelenik az első **10 legjobb eredmény** ez **resultstablereadprint modulban** található.

„Kilépés” Ezen lehetőség esetén egy **exit()** paranccsal leállítjuk a játékot.

### **test:**

Az említett függvény a **testquestionsandhelp.py központi file-ban** található. A főmenüben a „Játék indítása” opció választása esetén a kerül meghívásra. A függvény első felében a **readdata()** kerül meghívásra, itt a felhasználó által választott nehézségnek megfelelő lista kerül mentésre a **lista1** nevű változóban.

Az első 11 kérdés csak olyan nehézségű lehet, amely a kapott listában szerepel és a **questionlist** pedig megadja melyek ezek a nehézségek. Itt össze vannak gyűjtve az egyes nehézségi szintekhez kapcsolódó kérdések, melyek nem a legnehezebbek. Véletlenszerűen választja ki a program a kérdéseket oly módon, hogy az adott nehézséghez kapcsolódó legnehezebb kérdések ki lettek szűrve belőle a korábban említett a **lista1** és a **questionlist** összehasonlításával. Az utolsó 6 kérdés a 11.-től pedig pont az ellenkezőjét vizsgálja, ha nincs a nehézség **questionlist**-ben ami a könnyű kérdéseket összesíti csak akkor írja ki. A **questionlist**-tel való összehasonlításon kívül a korábbi kérdések egy listában tárolódnak és ellenőrzi ezt is a program, hogy ne forduljon elő ismétlés.

A legnehezebb „hard” lista esetén azért van a 14 is megadva a **questionlist**-ben, mivel a 15-ös nehézségű kérdésekből mindössze 3 db állt rendelkezésre és ismétlődés nélkül nem lehetne megoldani.

A **hpda** listára azért van szükség, hogy a program elindulását követően alap állapotba kerüljenek a korábban felhasznált segítségek. A **money** pedig egyelemű listában tárolja a nyereményt és 0-ra állítja a nyeremény aktuális nagyságát a játék kezdetekor.

1-10-ig a kérdések random kerülnek kiválasztásra, 2 paraméternek kell megfeleljen a véletlenszerűen választott kérdés: ne legyen az adott játékon belül ismétlődő, illetve a legnehezebb kérdéseknél legyen könnyebb. 11 feletti kérdéseknél ugyanez az első feltétel, csak ott a legnehezebb kérdések kerülnek megjelenítésre az adott nehézséghez kapcsolódó listából. A kérdés kiválasztása után meghívásra kerül a **questions()** függvény, amely a kérdések kiírását végzi.

#### questions:

Az említett függvény a **testquestionsandhelp.py** központi file-ban található. A test függvényből kerül meghívásra, paramétere (kérdés, az egyes kapcsolódó válaszlehetőségek, helyes válasz és a kérdés sorszáma, nyeremény nagysága. A **money** néven futó változó ebben a függvényben változik, amennyiben helyes a választ kap 1000Ft-al növekszik.

A **selectanss** válaszlehetőség alapvetően egy helytelen értékre van beállítva, amennyiben nem élünk segítséggel, ne egy üres értéket tároljon. A kérdés megjelenítését követően először megvizsgáljuk, nem-e **None** érték a visszatérési értéke. **None** érték úgy születhet, ha a kérdésdobozban a felhasználó vagy a **cancelled** vagy az „x”-et választotta (ilyenkor visszatér a játék a főmenübe) **main()** függvény meghívásával. Ez után pedig azt ellenőrizzük, hogy segítséget kért-e a játékos. Ha segítséggel szeretne élni a **help()** kerül meghívásra a szükséges paraméterek megadásával ezek *(kérdés,válaszok(4db), helyes válasz, a segítségeket tartalmazó lista)*. Ezt követően pedig a segítség eredménye megjelenik a kérdések felett, ez alapján tud választani a játékos. Amennyiben a felhasználó úgy kíván segítséggel élni, hogy már nem áll rendelkezésre úgy kattintásra eltűnik ez a lehetőség és megjelenik, hogy már nem élhet vele.

Helyes válasz esetén növekszik 1000Ft-al a nyeremény és egy újabb kérdéssel tér vissza a játék. A 16. kérdésre leadott helyes válasz, vagy bármelyikre leadott helytelen válasz esetén, bekérésre a kerül a felhasználó neve az eredmények mentéséhez és véget ér a játék. A **resultstablereadprint** modul **resultsappend()** függvénye által kerülnek egy külső text file-ba az adatok (**name,money,ido**) és ezt követően a visszalép a **main()**-be.

#### help:

Az említett függvény a **testquestionsandhelp.py** központi file-ban található. A **questions()** -ből kerül meghívásra. Az egyes segítségek elnevezése egy listában van tárolva. Az alábbi paramétereket kapja(kérd,a,b,c,d,hely,hpd) Attól függően, mennyi segítség áll rendelkezésre, úgy kerülnek kiírásra a segítségek. Az egyes segítségek a **hpda** listában vannak tárolva, amennyiben elhasználnunk belőle, úgy kikerül az adott segítség a listából a kapcsolódó változó is módosul és csak játék a újra indulását követően állnak vissza az adatok a játék elinduláshoz szükséges alapértelmezettre.

Attól függően, hogy a felhasználó a három segítség közül melyiket választotta a **helpoption** modul **phone()** -mint telefonos /**twooption()** -mint felezés / **spectators()** -mint közönség függvénye kerül meghívásra, ezeknek a függvényeknek csak a helyes megoldásra van szükségük paraméterként. Visszatérési értékük egy sztring vagy egy melyben a segítségnek megfelelő formában és valószínűséggel jelenik meg a helyes válasz.

#### main:

Az említett függvény a **testquestionsandhelp.py** központi file-ban található. A játék indulását követően innen kerül meghívásra a **menu()** függvény és indul a játék.

## Még nem kifejlesztett modulok:

### 1. readdatafromcsv

A **testquestionsandhelp.py** központi fájl **test()** függvényéből juthatunk ide, a felhasználó által választott nehézségnek megfelelő listával tér vissza.

#### 1.1. readdata:

Az említett függvény a **readdatafromcsv** file-ban található. A **kerdesek.csv** fájlból beolvassa a kérdéseket, nehézség alapján vizsgálva a **Questions** osztálynak megfelelően 3db listába tölti az elemeket. A „.csv” fájlban megadott nehézséget 3 csoportra bontva töltjük a könnyűeket a „**easy**” elnevezésű listába (0-5), a közepesek a „**medium**”-ba (5-10), illetve a nehéz a „**hard**” listába (10-15). A beolvasásnál megadott **with** függvény automatikusan kezeli az esetleges kivételeket. A felhasználó választásának megfelelően a egy **lista kerül továbbításra**, melyben a választott nehézségű kérdések kapnak helyet.

### 2. helption.py:

A **testquestionsandhelp.py** központi file **help()** függvényből juthatunk el ide, itt az egyes segítségnek megfelelően 3 db függvényt találunk, mindhárom a helyes megoldást kapja meg paraméterként és visszatérési értékük egy sztring vagy egy lista.

#### 2.1. phone:

Az említett függvény a **helption.py** file-ban található. Röviden egy listába rakja a 3 helytelen megoldást (súlyuk összesen 20%) és az egy helyes megoldást (ennek súlya 80% -os). Ebből választ random módon a program és az a **betűjel** lesz a visszatérési értéke.

#### 2.2. half:

Az említett függvény a **helption.py** file-ban található. Röviden egy listába rakja a helyes és egy random választott nem helyes megoldást ezt utána sorba rendezi és ez a **2 elemű lista lesz a visszatérési értéke**, ismétlődés nélkül. Tehát itt 50%-os valószínűséggel kap helyes választ a játékos.

#### 2.3. spectators:

Az említett függvény a **helption.py** file-ban található. A program 75-90% közötti valószínűséggel jeleníti meg a felsorolásban a helyes megoldást, ami jelen példában azt jelöli a közönség mekkora része szavazott erre, a többi a lehetőség random valószínűséggel jelenik meg úgy, hogy mindig a 100% és az aktuális összes maradékból random választ egy számot a program.

**Visszatérési értéke egy rendezett sztring**, ebből a felhasználó láthatja a közönség mekkora része szavazott az adott válaszlehetőségekre. („X”jelekkel 5%=1 db „X”) megjelenik az eloszlás is a sztringben.

### 3. resultstablereadprint.py:

#### 3.1. resultsappend:

A **testquestionsandhelp.py** központi file **test()** függvényből kerül meghívásra miután a felhasználó megadta a nevét játék vége után a játékban elért eredmények pontosvesszővel tagolva egy külső „**dicsoseg.txt**” text file-ban, eredményenként új sorban pontosvesszővel tagolva tárolódnak el.

#### 3.2. resultlist:

A **testquestionsandhelp.py** központi file **main()** függvényből a „**Dicsőségtábla**” lehetőséget választva juthatunk el ide. A függvény addig, amíg üres sort nem talál beolvassa az adatokat a „**dicsoseg.txt**” text fájlból listába tölti. Ezek után először rendezi a nyeremény nagysága szerint utána pedig amennyiben két azonos nyeremény van, de egyik rövidebb idő alatt született, még egyszer rendez idő alapján is. Az így kapott lista 10 legjobb eredményét egy sztringbe tölti és ez követően ez kerül kiírásra.

## Megjelenítés:

### easygui:

Az említett file a **easygui.py** a **játék mappájában** található. A file-t a <http://easygui.sourceforge.net/> weboldaltól szereztem be és az itt található **tutorial** alapján dolgoztam. Az **easygui.py** fájlban kívül nem igényel semmit.

Az alábbi megjelenítési formákat használtam:

```
selectmenu = easygui.buttonbox("Legyen Ön is milliomos",image = "1-kép.png",choices  
= ["\n Játék indítása\n", "\n Dicsőségtábla \n", "\n Kilépés\n"])
```

A **buttonbox** függvény került meghívásra az **easygui.py** fájlból, az első megadott paraméter a szövegdobozban megjelenít egy feliratot, a következő egy képet, a legutolsó pedig gombokon a választási lehetőségeket. A felhasználó által választott lehetőség lesz az input értéke.

```
selectans = easygui.choicebox(qstion,title=("Eddigi nyeresége: {}  
Ft".format(moneyy[0])),choices = [str(optiona), str(optionb), str(optionc),str(optiond)])
```

A **choicebox** függvény került meghívásra az **easygui.py** fájlból, az első megadott paraméter egy adatot jelenít meg a szövegdobozban a következő pedig a fejlécben megjelenő adatokat adja, mint láthatjuk itt egy változóval a nyereséget, choices pedig lista szerűen ABC sorba rendezve megjeleníti a választási lehetőségeket. Az egyes opcióra kattintással adjuk meg az input értékét. A választott lehetőség adja meg a *selectans* változó értékét

```
nev=easygui.enterbox(msg = "Gratulálunk megnyerte a főnyereséget! A  
nyeresége:{} Idő {:0.5}mp // Kérem adja meg a nevét a  
mentéshez:".format(moneyy[0],ido),title=("Eddigi nyeresége: {} Ft".format(money)),  
default="", image = "2-jtk.png",)
```

Az **enterbox** függvény került meghívásra az **easygui.py** fájlból, az első megadott paraméter a korábbiakhoz hasonlóan adatot jelenít meg a szövegdobozban a következő pedig a fejlécben megjelenő adatokat adja, mint láthatjuk itt egy változóval az eddigi nyereséget, végül egy képet jelenít meg. A **nev** változó pedig jelen esetben a szövegdobozba a felhasználó által begépelt adatot fogja tartalmazni.