

Projekts

Spēle “9 dzīvības”

1. Ir dots programmas kods. Pārraksti kodu ievērojot visas atkāpes.
2. Rakstot kodu, raksti komentārus par notiekošo.
3. Labo visas kļūdas lai programma darbotos pareizi.

```
#Programma izvada uz ekrāna vārdu, kuram burtu vietā ir jautājumu zīmes.
Ja tu uzminēsi burtu, programma, to atvērs. Ja tu zini visu vārdu, ievadi
to. Spēle beigsies, kad uzminēsi vārdu vai beigsies visas dzīvības
print ("Spēle: '9 dzīvības'")
import random
dzivibas = 9
vardi = ['vista', 'kaķis', 'ābols', 'čūska', 'varde', 'spēle', 'uguns',
'karte', 'kuģis']
secret_vards = random.choice(vardi)
minejums = list('?????')
sirds_symbols = u'\u2665' #ASCII Unicode character
uzminets_vards_pareizi = False
def update_atrisinajums(minets_burts, secret_vards, atrisinajums ):
    index = 0
    while index < len(secret_vards):
        if minets_burts == secret_vards[index]:
            atrisinajums[index] = minets_burts
            index = index + 1
    while dzivibas > 0:
        print(minejums)
        print ( 'Dzīvības ' + sirds_symbols * dzivibas )
        atbilde = input('Uzmini burtu vai visu vārdu: ')
        if atbilde == secret_vards:
            uzminets_vards_pareizi = True
            break
        if atbilde in secret_vards:
            update_atrisinajums(atbilde, secret_vards, minejums)
        else:
            print( 'Nepareizi! Jūs pazaudējat dzīvību!')
            dzivibas = dzivibas - 1
        if uzminets_vards_pareizi:
            print( 'Pareizi! Vārds bija: ' \
+ secret_vards)
        else:
            print( 'Tu zaudēji!:( Vārds bija: ' \
+ secret_vards)
```

4. Apskaties, kas ir ASCII Unicode characters, vai iespējams tos izmainīt pret citiem.
5. Papildini spēles sākumu, uzprasot lietotājam vai nepieciešams paskaidrot spēles noteikumus, ja “jā”, tad paskaidro, ja “nē”, tad novel veiksmi.
6. Papildini programmu, lai lietotājs varētu izvēlēties grūtības pakāpi, jo augstāka grūtības pakāpe, jo mazāk dzīvību.