

Python 2025.05. Kihívás

Az interneten számos sportkihívással találkozhatunk. Ezek általában egy adott időszakra tűznek ki valamilyen elérendő célt, ezzel is mozgásra ösztönözve az embereket. Ebben a feladatban egy heti mozgáskihívás eredményeit kell kiértékelnie!

A kihívásban a heti mozgást egy applikáció segítségével kellett rögzíteni és a hét végén beküldeni. A kihívást a következő mozgásformák segítségével lehetett teljesíteni: úszás, gyaloglás, futás, kerékpározás. A kihívás célja 40 km elérése volt. Az applikáció rögzítette a heti mozgást, a felhasználó pedig a hét végén beküldte a rögzített teljesítményt.

A beküldött érték a mozgásforma betűjelét tartalmazza a megtett távolságoknak megfelelően. Az alábbi táblázatban láthatóak a betűjelek és a hozzájuk tartozó távolságértékek:

Mozgásforma	Kód	Kódhoz tartozó távolság
Úszás	U	1 km
Gyaloglás	G	1 km
Futás	F	2 km
Kerékpározás	K	10 km

Az alábbi példa egy felhasználó heti aktivitását mutatja:

```
FFFGGGUUUFFFGGKKK
```

A felhasználó a héten a következő tevékenységeket végezte:

FFF – 6 km futás, GGG – 3 km gyaloglás, UUU – 3 km úszás, FFF – 6 km futás, GG – 2 km gyaloglás, KKK – 30 km kerékpározás. Ezzel 50 kilométert teljesített.

Készítsen programot, amely kiértékeli a beküldött aktivitást! A program forráskódját mentse *kihivas* néven! A program megírásakor a felhasználó által megadott adatok helyességét, érvényességét nem kell ellenőriznie, és feltételezheti, hogy a rendelkezésre álló adatok a leírtaknak megfelelnek.

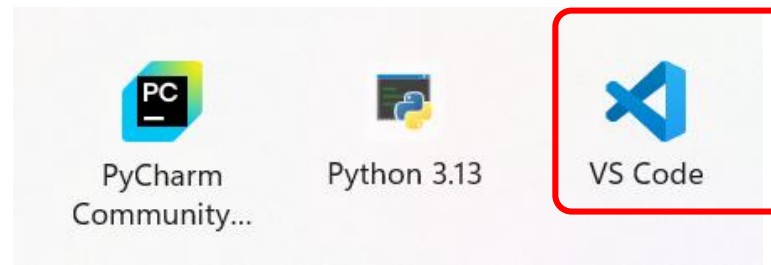
A feladat leírása

A képernyőre írást igénylő részfeladatok esetén az ékezetmentes kiírás is elfogadott. A mintához tartalmában hasonlóan – írja ki a képernyőre a feladat sorszámát (például: 2. feladat), valamint utaljon a kiírt tartalomra is!

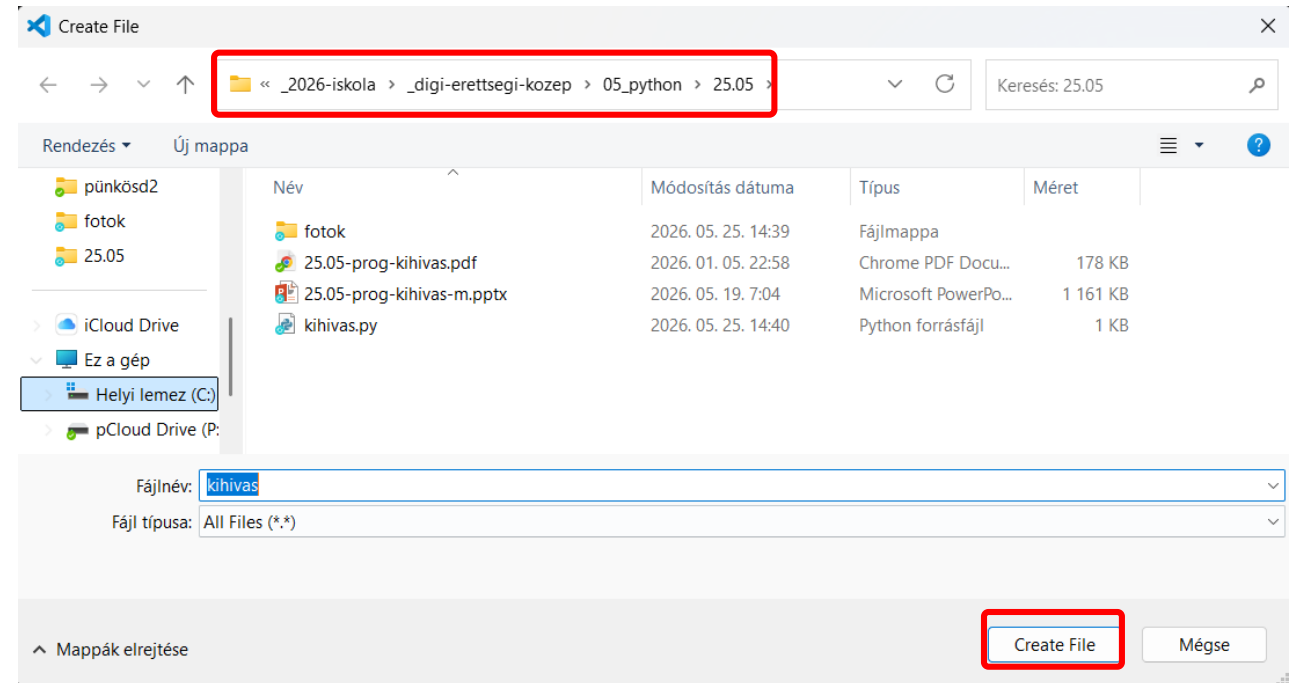
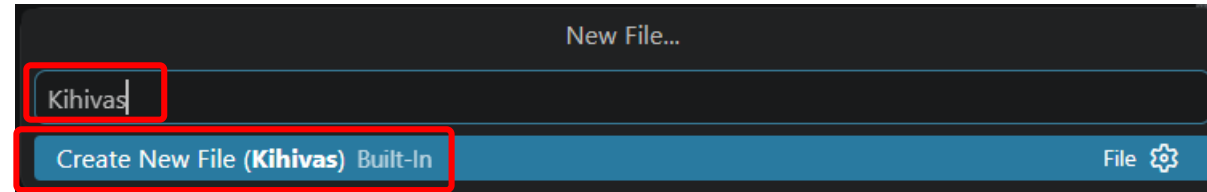
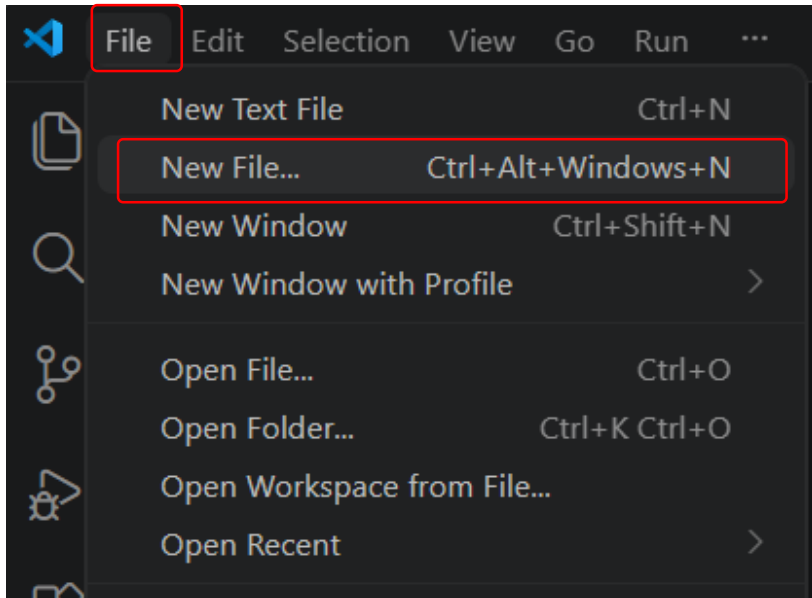
1. Kérje be és tárolja el a felhasználó heti aktivitását! Feltételezheti, hogy a megadott sorozat hossza 250 karakternél rövidebb.
2. Számítsa ki és a mintának megfelelően jelenítse meg a felhasználó aktivitását, azaz a héten megtett távolságok összegét!
3. Amennyiben a felhasználó mindegyik mozgásformát üzte az adott héten, akkor a heti teljesítményéhez 10 km pluszt kap! Ha a felhasználó teljesítette ezt a feltételt, írassa ki, hogy „Bravó! Jutalma még 10 km.”, különben pedig a „Nem jár jutalom.” üzenetet jelenítse meg!
4. Írassa ki a képernyőre a felhasználó által gyűjtött kilométerek számát, amely a megtett heti távolságérték és a kapott jutalomkilométerek összege! Ha a gyűjtött kilométerek eléri a heti kihívásnak megfelelő 40 km-t, akkor a „Gratulálok, kihívás teljesítve!” üzenetet jelenítse meg, a minta szerint! Amennyiben nem teljesítette a kitűzött célt, a „Legközelebb sikerül!” üzenetet jelenítse meg!

```
1. feladat
Adja meg az aktivitását: FFFGGGUUUFFFFGGKKK
2. feladat
Az elért távolság: 50 km.
3. feladat
Bravó! Jutalma még 10 km.
4. feladat
Eredménye: 60 km. Gratulálok, kihívás teljesítve!
```

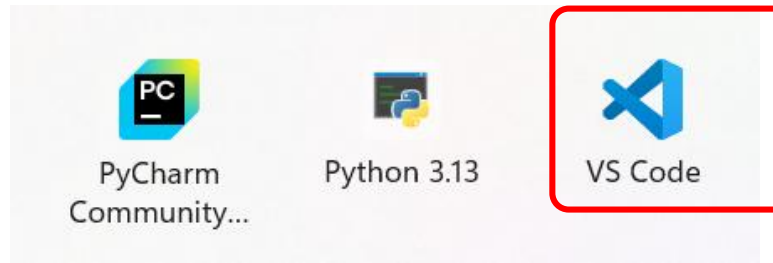
VS Code



File -> New Project -> kihivas-> Create New File-> Mentés helye -> Create File

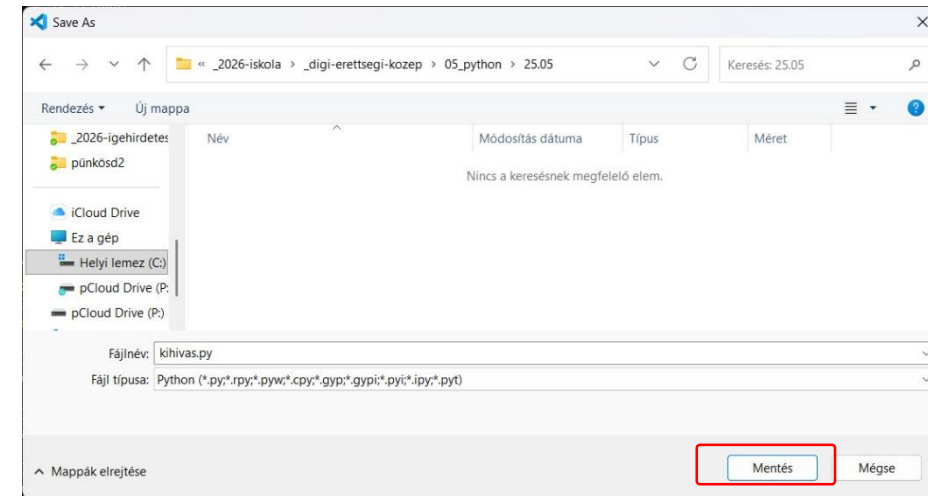
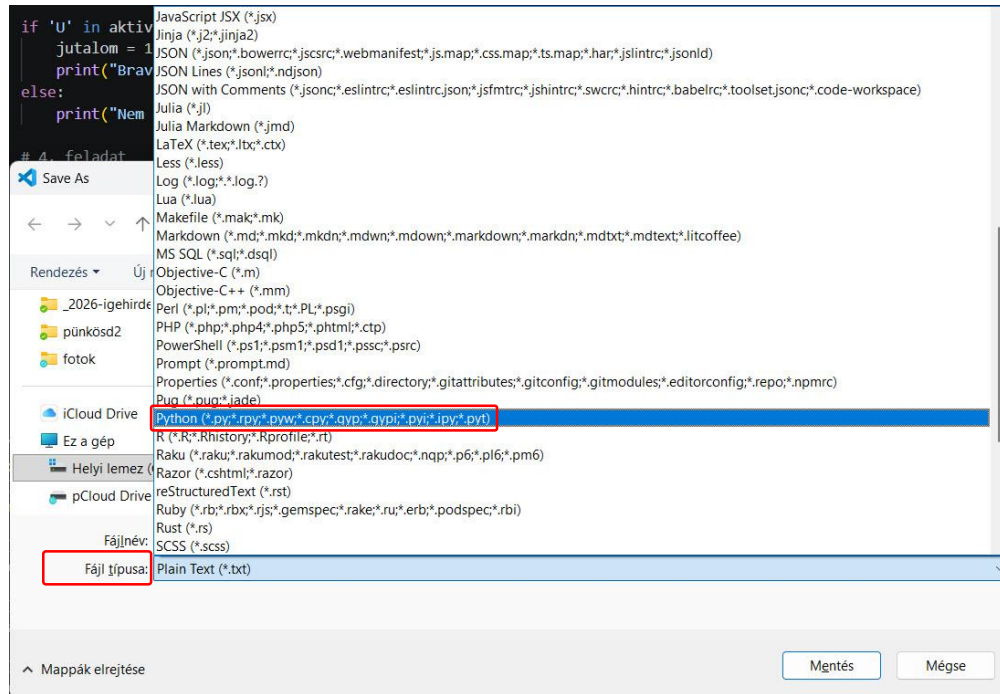
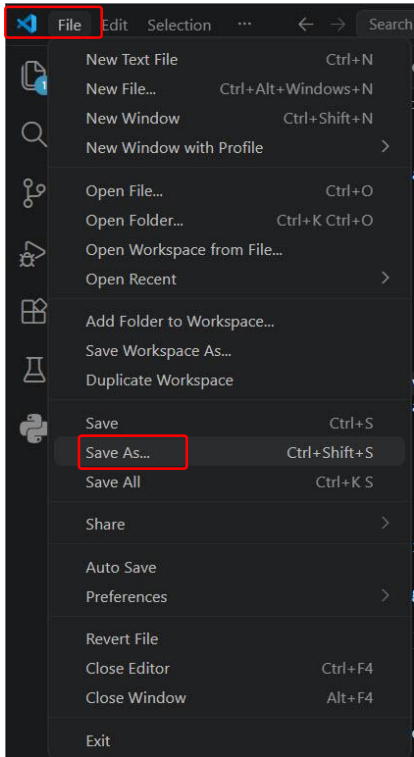


VS Code

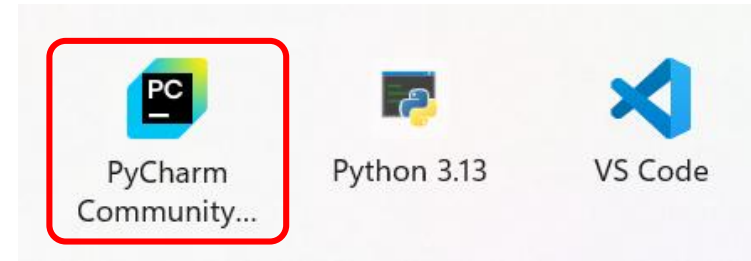


File -> New Project -> befozes-> Create New File-> Mentés helye -> Create File

File -> Save As... -> Fájl típusa: -> Python -> Mentés

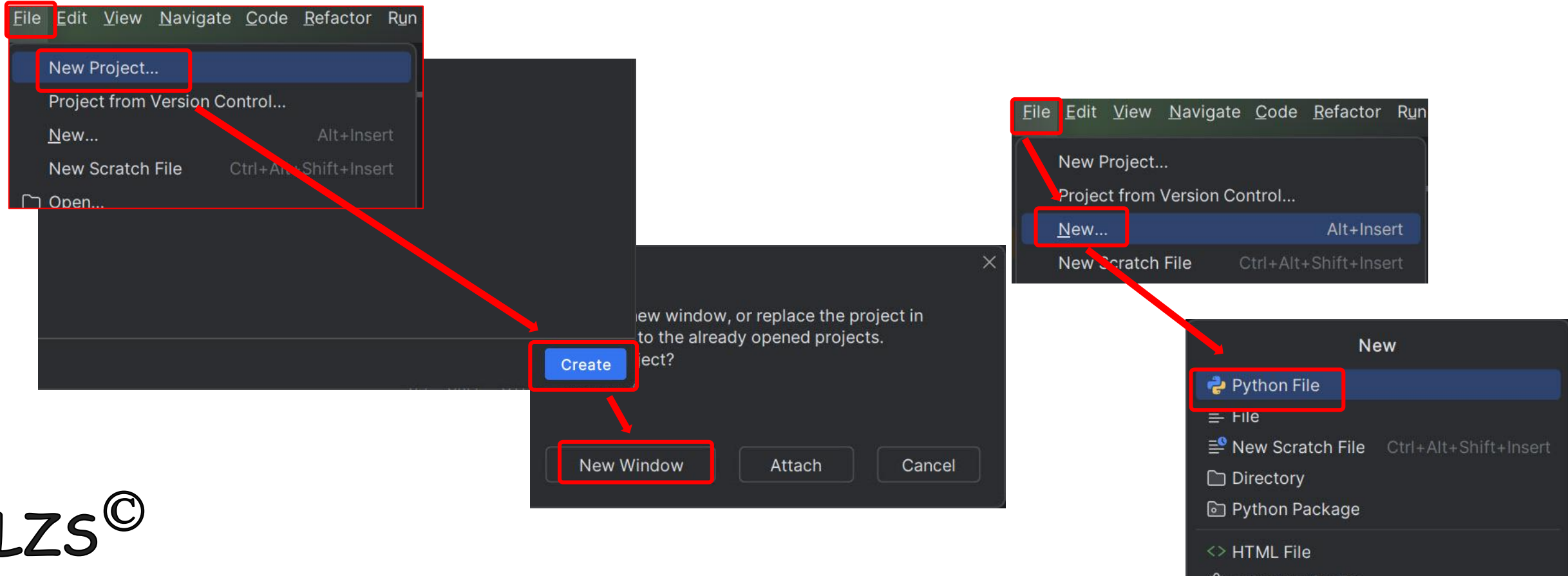


PyCharm

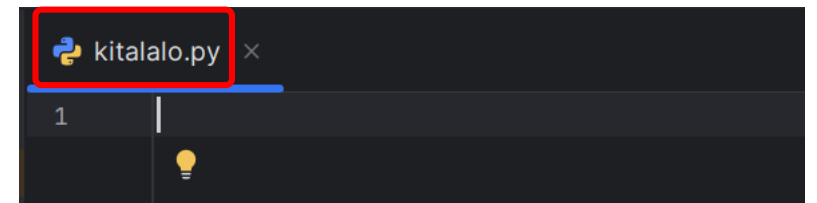
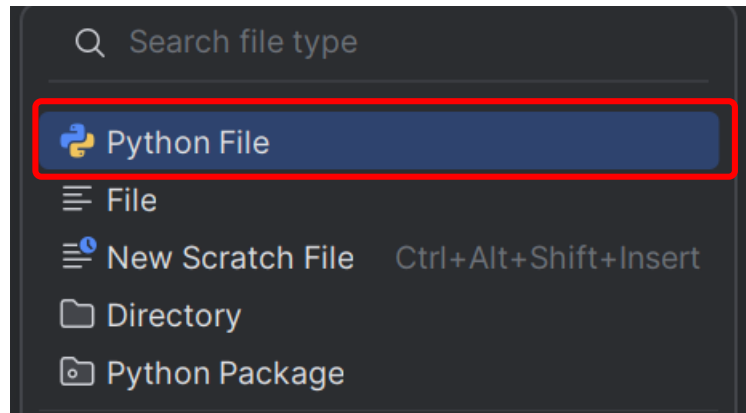
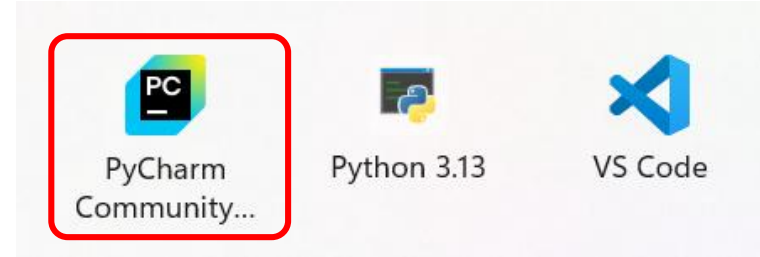
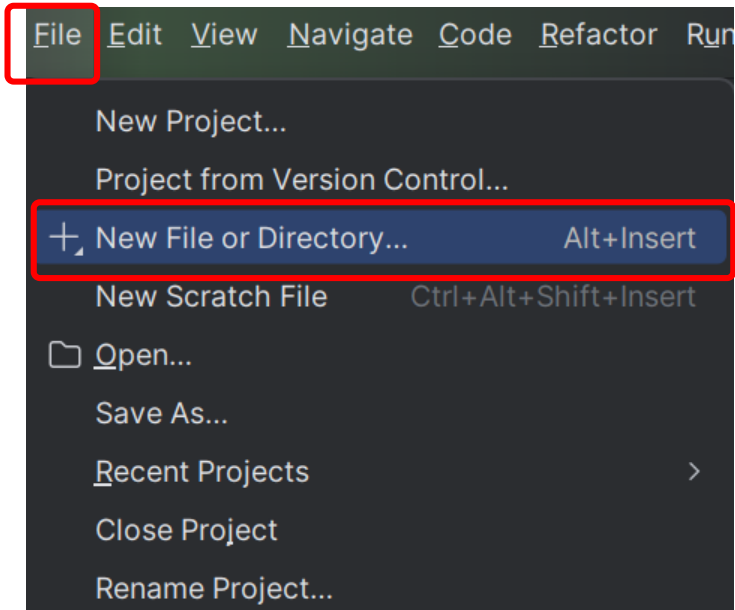


Suliban:

File -> New Project -> Create -> New Windows -> File -> New -> Python File



PyCharm



Az elkészített program

```
C: > hp-500 > _2026-iskola > _digi-erettsegi-kozep > 05_python > 25.05 > 🚀 kihivas.py > ...
1 # 1. feladat
2 print("1. feladat")
3 aktivitas = input("Adja meg az aktivitását: ")
4
5 # 2. feladat
6 print("\n2. feladat")
7 u_tav = aktivitas.count('U') * 1
8 g_tav = aktivitas.count('G') * 1
9 f_tav = aktivitas.count('F') * 2
10 k_tav = aktivitas.count('K') * 10
11
12 alap_tavolsag = u_tav + g_tav + f_tav + k_tav
13 print(f"Az elért távolság: {alap_tavolsag} km.")
14
15 # 3. feladat
16 print("\n3. feladat")
17 jutalom = 0
18
19 if 'U' in aktivitas and 'G' in aktivitas and 'F' in aktivitas and 'K' in aktivitas:
20     jutalom = 10
21     print("Bravo! Jutalma még 10 km.")
22 else:
23     print("Nem jár jutalom.")
24
25 # 4. feladat
26 print("\n4. feladat")
27 vegleges_tavolsag = alap_tavolsag + jutalom
28
29 if vegleges_tavolsag >= 40:
30     print(f"Eredménye: {vegleges_tavolsag} km. Gratulálok, kihívás teljesítve!")
31 else:
32     print(f"Eredménye: {vegleges_tavolsag} km. Legközelebb sikerül!")
33
```

```
1. feladat
Adja meg az aktivitását: FFFGGGUUUFFFFGGKKK
2. feladat
Az elért távolság: 50 km.
3. feladat
Bravó! Jutalma még 10 km.
4. feladat
Eredménye: 60 km. Gratulálok, kihívás teljesítve!
```

1. feladat: Adatbekérés

Mit csinál?

A `print()` parancs kiírja a képernyőre a szöveget.

Az `input()` parancs pedig várja, hogy a felhasználó gépeljen be valamit (a megadott karaktersorozatot), és nyomjon egy Entert.

Amit begépelte, azt beletesszük az `aktivitas` nevű "dobozba" (változóba).

Gyakori hiba:

Sokan elfelejtik, hogy a szövegeket (úgynevezett stringeket) idézőjelek `" "` vagy aposztrófok `' '` közé kell tenni a programkódban.

Ha lebegyod, a gép hibát jelez.

```
1 # 1. feladat
2 print("1. feladat")
3 aktivitas = input("Adja meg az aktivitását: ")
4
```

2. feladat: Távolság kiszámítása

A `\n` a `print` belsejében csak egy sortörést jelent (szebb a kimenet).

Mivel tudjuk, hogy $U=1$, $G=1$, $F=2$, $K=10$ km, a Python számolja meg a betűket!

A `.count('U')` megszámolja az összes nagy 'U' betűt a megadott szövegben.

Ezt beszorozzuk a km-rel, és beletesszük új változókba. Végül ezeket összeadjuk.

Az f-string: A kiírásnál a `print(f"...")` azt jelenti, hogy a kapcsos zárójelek `{ }` közötti változók helyére helyettesítse be a bennük lévő számot.

Gyakori hiba: A Python kis- és nagybetű érzékeny! Továbbá hiba lehet, ha elfelejtet a matematikai műveletek jeleit (pl. a `*` a szorzás jele).

```
6 print("\n2. feladat")
7 u_tav = aktivitas.count('U') * 1
8 g_tav = aktivitas.count('G') * 1
9 f_tav = aktivitas.count('F') * 2
10 k_tav = aktivitas.count('K') * 10
11
12 alap_tavolsag = u_tav + g_tav + f_tav + k_tav
13 print(f"Az elért távolság: {alap_tavolsag} km.")
14
```

3. feladat: A bónusz megállapítása

Létrehozunk egy jutalom nevű dobozt, amibe alapból 0-t teszünk.

Az if (ha): egy feltételt vizsgálunk.

Az in -el megkérdezzük: "Benne van az 'U' betű az aktivitásban?". Az and (és) összekötjük a feltételeket, mert a feladat szerint mindegyik mozgásformát végezni kellett. Ha igaz, a jutalom dobozba 10-et teszünk, és kiírjuk print a dicséretet.

Ha nem teljesül (else, azaz egyébként), kiírjuk, hogy nem jár.

Gyakori hiba: A kettőspont (:) lefelejtése az if és az else sorok végéről nagyon gyakori! Szintén tipikus hiba az indentáció (behúzás) hiánya. A Python abból tudja, hogy mi tartozik az if belsejébe, hogy a sorok beljebb vannak húzva (általában 4 szóközzel vagy egy Tab-bal). Ha ezt elrontod, a program összeomlik.

```
16 print("\n3. feladat")
17 jutalom = 0
18
19 if 'U' in aktivitas and 'G' in aktivitas and 'F' in aktivitas and 'K' in aktivitas:
20     jutalom = 10
21     print("Bravo! Jutalma még 10 km.")
22 else:
23     print("Nem jár jutalom.")
24
```

4. feladat: Végeredmény

Mit csinál?

Összeadjuk az alaptávolságot és a jutalmat. A kihívás célja 40 km volt. Újra használjuk az if-et: megvizsgáljuk, hogy a **végső távolság nagyobb vagy egyenlő-e (>=), mint 40**. Ennek megfelelően írjuk ki a záró üzeneteket.

Gyakori hiba:

Sokan csak > (nagyobb) jelet használnak, ami hiba, hiszen ha valaki pont 40 km-t ér el, arra a > 40 feltétel hamis lesz, pedig azzal is teljesítette a kihívást. Ezért kell a >= (nagyobb vagy egyenlő).

```
26 print("\n4. feladat")
27 vegleges_tavolsag = alap_tavolsag + jutalom
28
29 if vegleges_tavolsag >= 40:
30     print(f"Eredménye: {vegleges_tavolsag} km. Gratulálok, kihívás teljesítve!")
31 else:
32     print(f"Eredménye: {vegleges_tavolsag} km. Legközelebb sikerül!")
33
```