**Informatika és távközlés ágazat - ágazati alapvizsga**

***GYAKORLATI VIZSGA  
Weboldalak kódolása, programozás,  
hálózatok gyakorlat***

**B) Programozás Pythonban feladatrész**

**A feladatrész feladataira vonatkozó általános elvárások:**

* Az Ön feladata az alábbiakban olvasható leírás alapján három Python-program elkészítése, amelyekre összesen kb. 60 perc áll rendelkezésre.
* A programokat a vizsgaszervező által kijelölt helyre és a feladatleírásban szereplő néven kell mentenie.
* A programok elkészítése során a felhasználó által megadott adatok helyességét nem kell ellenőriznie – ha például a program egy 1 és 5 közé eső szám megadását kéri a felhasználótól, akkor feltételezheti, hogy a felhasználó számot, és a feltételeknek megfelelő számot ad meg.
* Törekedjen arra, hogy a tanult programozási elveknek megfelelő adatszerkezeteket, vezérlési szerkezeteket alkalmazzon!
* Munkáját mentse rendszeresen!

**1. feladat**

Írjon programot **A001.py** néven!

A program kérjen be két számot a felhasználótól, majd írja ki, hogy melyik a kisebb!

A program addig kérje be az adatokat, amíg a szám megadásakor **üres** bemenetet nem kap!

**A program üzeneteinek megfogalmazásában kövesse az alábbi példát!** Azokat a részeket, amelyeket a felhasználó gépel be, a „Futási példa:” mintában vastagított betűkkel emeltük ki.

Futási példa

Adj meg egy számot! ***11***

Adj meg egy másik számot! ***17***

* 1. kisebb érték 11

Adj meg egy számot! 1***8***

Adj meg egy másik számot! 1***2***

A kisebb érték: 12

Adj meg egy számot! ***11***

Adj meg egy másik számot! ***11***

A két szám egyenlő.

Pontozás – minden teljesülő feltétel egy-egy pontot ér

1. Létrehoz programot kisebb.py néven, a program hibaüzenet nélkül lefut.
2. Bekéri a felhasználótól az számokat, és tárolja.
3. A bekért számokat szám típusúvá alakítja.
4. Ciklus szervez üres bemenetig
5. Elágazást használ a különböző esetek kezelésére.
6. Ha a két szám nem egyenlő, helyesen állapítja meg és írja ki, hogy melyik a kisebb.
7. Helyesen állapítja meg, és írja ki, ha a két szám egyenlő.
8. A kiírt üzenetek helyesek (pl.: nincs benne elgépelés, helyesen jelennek meg a szóközök).

**2. feladat:**

Írjon programot **A002.py** néven!

Írjon függvényt a négyzet és a kör területének kiszámítására. A számolást math modul segítségével valósítsa meg!

Függvényt használjon a programjában! A függvény bemenő paramétere az alakzat típusa és a bekért szám legyen, a visszatérési érték pedig a kiszámított terület.

A program kérdezgesse a adadatokat, amíg üres bemenetet nem kap az alakzat típusánál! Ilyen akkor történik, ha a felhasználó egyszerűen Entert nyom, anélkül hogy bármit is begépelne

**A program üzeneteinek megfogalmazásában kövesse az alábbi példát!** Azokat a részeket, amiket a felhasználó gépel be, a mintában vastagított és döntött betűkkel emeltük ki.

(N)égyzet vagy a (K)ör területét szeretné kiszámolni: ***N***

Az oldalak vagy a kör sugara : ***6***

Az alakzat területe:

N)égyzet vagy a (K)ör területét szeretné kiszámolni:

Pontozás

1. Létrehozta a programot, a program hibaüzenet nélkül lefut.
2. Bekér egy számot és tárolja.
3. Bekéri az adatokat.
4. Egy esetben helyesen jelenít meg üzenetet. Az üzenet az alakzat nevét is feltünteti.
5. Ciklust szervez az adatok bekérésére, illetve az eredmények kiírtatására.
6. A ciklus futása véget ér, ha az alakzat megadásakor üres bemenetet kap a program és ilyenkor nem kéri be a többi adatot.
7. Függvényt hozott létre az eredmények kiszámolására (kör és négyzet kerülete).
8. A függvény paramétere a bekért szám és az alakzat típusa.
9. A függvényt helyesen hívja.
10. A függvény visszatérési értéke a kör vagy a négyzet kerülete.
11. A függvényhívás a ciklusmagba kerül.
12. A függvény visszatérési értéke alapján a főprogram (vagy az annak megfelelő függvény) írja ki a közösségi szolgálat meglétét.

**3. feladat:**

Az elkészítendő program egy autókereskedés adatait tárolja objektumokban. Az adatokat a kereskedés.txt fájlba találja. Írjon programot **A003.py** néven!

Az adataok a következő tulajdonsággal rendelkezzenek: az autó típusa (márkanév), melyik évben gyártották, és a bruttó árát.

1. Olvassa be az kereskedes.txt fájlból az adatokat és tárolja el azokat! Az adatok tárolására szolgáló objektumok alapját képező osztályt hozza létre.
2. Állapítsa meg, hogy hány autó szerepel az adatbázisban.
3. Írja ki a képernyőre az eltárolt adatokat! (típus, év, ár)
4. Állapítsa meg az autók átlagárát és írja ki a képernyőre.
5. Írjon ki véletlenszerűen egy tanuló adatait (típus, év, ár).
6. Határozza meg az autók átlagárát.
7. Határozza meg mennyibe kerül a legdrágább autó és írja ki a képernyőre.
8. Határozza meg a legdrágább autó típusát és írja ki a képernyőre! Feltételezheti, hogy nincsennek egyező értékek.
9. Határozza meg mennyibe kerül a leolcsóbb autó és írja ki a képernyőre.
10. Határozza meg a legolcsóbb autó típusát és írja ki a képernyőre! Feltételezheti, hogy nincsennek egyező értékek.
11. Írja ki az eltárolt autók típusait a képernyőre és írja be a „tipusok.txt” szövegfájlba!

**A program üzeneteinek megfogalmazásában kövesse az alábbi példát**! Azokat a részeket, amiket a felhasználó gépel be, a mintában vastagított betűkkel emeltük ki.

**Futási példa:**

Adatok kiíratása:

Suzuki: 2008 év, 5600000 Ft

Opel: 2008 év, 4500000 Ft

Renault: 2009 év 7600000 Ft

Honda: 2006 év, 4510000 Ft

Suzuki: 2008 év, 5600123 Ft

Suzuki: 2009 év, 6400000

Toyota: 2009 év, 8900000 Ft

Skoda: 2008 év, 4570000 Ft

Seat Ibiza: 2011 év, 4900000 Ft

Az adatbázisban lévő tanulók száma: 9 db

Az autók átlagára:

Egy autó véletlenszerűen:

Skoda: 2008 év, 4570000 Ft

A legdrágább autó adatai: 8900000 Ft

A legolcsóbb autó: 4500000 Ft

A legdrágább autó adatai:

Toyota: 2009 év, 8900000 Ft

A legolcsóbb autó adatai:

Opel: 2008 év, 4500000 Ft

Pontozás:

1. Létrehozta a programot az adott néven, a program hibaüzenet nélkül lefut.
2. Megnyitja a fájlt és legalább egy sort beolvas a fájlból.
3. Az összes adatot beolvassa a fájlból.
4. Létrehozza a megfelelő adatstruktúrát az adatok tárolásához.
5. Legalább egy esteben megfelelően eltárolja a beolvasott adatot.
6. Az összes adatot eltárolja a későbbi feladatokban használható adatstruktúrába.
7. Az autók k számának meghatározása és kiírásai.
8. Az eltárolt adatok minta szerinti kiírása.
9. Az átlagár meghatározása és kiírása.
10. Random modul importálása, és egy autó adatainak véletlenszerű kiírása.
11. A legdrágább autó meghatározása és kiíratása a képernyőre.
12. A legdrágább autó adatainak kiírása a képernyőre.
13. A legolcsóbb autó meghatározása és kiíratása a képernyőre.
14. A legolcsóbb autó adatainak kiírása a képernyőre.
15. Létrehozza az adott fájlt.
16. Legalább egy adatot a minta szerint ír a fájlba.
17. Az összes adatot a minta szerint írja a fájlba.
18. Kiírt üzenetek helyesek (pl.: nincs benne elgépelés, helyesen jelennek meg a szóközök). Egy hiba esetén még a pont megadható.