

8. Tervezési minták I. (Testek és Lények)

Határidő ápr 30, 23:59

Pont 10

Kérdések 10

Időkorlát Nincs

Engedélyezett próbálkozások 5

[Kvíz kitöltése újra](#)

Próbálkozások naplója

	Próbálkozás	Idő	Eredmény
MEGTARTOTT	3. próbálkozás	2 perc	8 az összesen elérhető 10 pontból
LEGUTOLSÓ	3. próbálkozás	2 perc	8 az összesen elérhető 10 pontból
	2. próbálkozás	4 perc	5 az összesen elérhető 10 pontból
	1. próbálkozás	4 perc	4 az összesen elérhető 10 pontból

⚠ A helyes válaszok el vannak rejtve.

Ezen próbálkozás eredménye: **8** az összesen elérhető 10 pontból

Beadva ekkor: ápr 23, 13:54

Ez a próbálkozás ennyi időt vett igénybe: 2 perc

1. kérdés

1 / 1 pont

Hol hivatkozhatunk egy osztály osztályszintű tagjaira?

- ☐ Csak az adott osztály publikus vagy védett osztályszintű metódusaiban.
- ☐ Csak az adott osztály osztályszintű metódusaiban.
- ☐ Az adott osztályban bárhol, az osztályon kívül csak akkor, ha az osztály publikus.
- ☒ Bárhol, ahol látható.

Helytelen

2. kérdés**0 / 1 pont**

Mikor inicializálódnak az osztályszintű adattagok?

- ☐ Az osztályt tartalmazó program végrehajtásának kezdetén.
- ☐ Az osztályra hivatkozó programrész végrehajtásának kezdetén.
- ☐ Amikor az osztályból először példányosítunk egy objektumot.
- ☒ Ezt a feladatot az osztályszintű konstruktor végzi.

3. kérdés**1 / 1 pont**

Mi a sablonfüggvény tervezési minta lényege?

- ☒ Egy alosztály örökölt metódusának működése ugyanazon ősosztálytól örökölt, de az alosztályban felüldefiniált más metódusoktól is függ.
- ☐ Olyan sablon (vagy C# nyelven generikus) osztály definiálása, amelynek egyik sablon-paramétere egy metódus.
- ☐ Egy sablon (vagy C# nyelven generikus) osztály metódusainak működése a sablon paramétereitől függ.
- ☐ Egy ősosztály virtuálisként megjelölt metódusait annak alosztályaiban felüldefiniálhatjuk, hogy érvényesüljön a dinamikus altípusos polimorfizmus.

4. kérdés**1 / 1 pont**

Milyen célt szolgál az egyke (singleton) tervezési minta?

- ☐ Memória szivárgás megszüntetése.
- ☐ Kód-redundancia megszüntetése.
- ☐ Futási idő javítása.
- ☒ Memória pazarlás mérséklése.

5. kérdés

1 / 1 pont

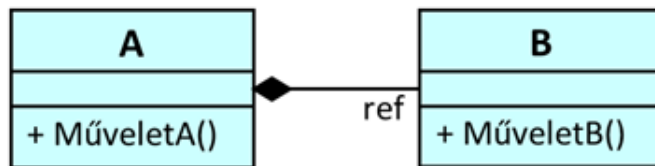
Milyen szoftvertervezési célt fejez ki az Open-Closed elv?

- ☐ A fejlesztőknek az alkalmazások készítésekor nyitottnak kell lenni a felhasználói igények befogadására, de miután a specifikációt rögzítették, nem szabad teret adni a tervezés és a megvalósítás során a specifikációtól eltérő felhasználói igényeknek.
- ☐ A perzisztenciához használt háttérfájlokat mindig meg kell nyitni a használat előtt, és le kell zárni azt követően.
- ☐ Azt, hogy az adatbázisok kulcsát meg kell őrizni.
- ☒ Egy szoftver későbbi módosítása során a már meglévő osztályokat, modulokat, metódusokat elég legyen csak újakkal kiegészíteni, változtatni ne kelljen.

6. kérdés

1 / 1 pont

Milyen célt szolgálhat az alábbi osztálydiagram?



Olyan objektumokat tudunk az A osztályból példányosítani, amelyekre a MűveletA() és a MűveletB() egyaránt hívható.



Olyan objektumot tudunk az A osztályból példányosítani, amelyek tartalmazza a B osztály egy objektumának hivatkozását, hogy a B osztály metódusait használhassa.



Olyan A típusú objektumok hozhatók létre, amelyeket a B típusú objektumok használhatnak fel.



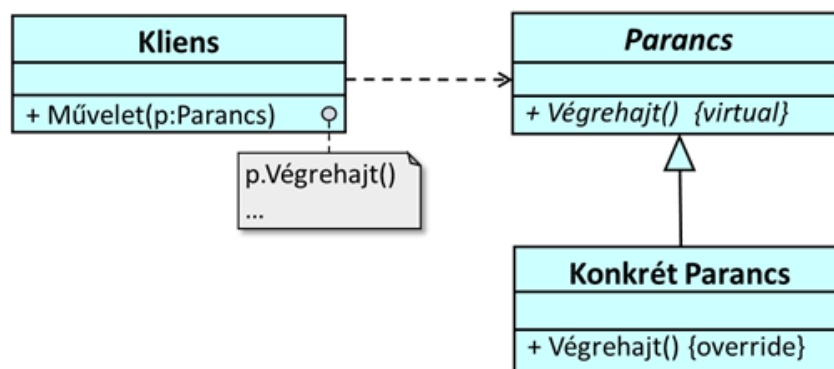
Olyan B típusú objektumok hozhatók létre, amelyek öröklik a MűveletA() metódust.

Helytelen

7. kérdés

0 / 1 pont

Mi olvasható ki az alábbi tervből?



A Kliens típusú objektumok akár a Parancs osztály, akár az abból származtatott KonkrétParancs osztály Végrehajt() műveletét használhatják.



Egy KonkrétParancs objektum Végrehajt() metódusa felhasználja egy Kliens objektum Művelet() metódusát.



Minden Kliens típusú objektum kapcsolatban áll legalább egy Parancsból származtatott osztály objektumával.



A Kliens típusú objektumok művelete a paraméterként megkapott parancs objektumtól függ.

8. kérdés

1 / 1 pont

Mi a stratégia tervezési minta lényege?



Olyan osztályokat definiál, amelyekből legfeljebb egy példány hozható létre.



Biztosítja, hogy egy metódus részét alkotó tevékenység rugalmasan megválasztható, lecserélhető legyen.



Segítségével megváltoztatható egy objektum szerkezete.



Hasonló tevékenységeknek származtatási hierarchiáját jelöli ki.

9. kérdés

1 / 1 pont

Mi az interfész szegregáció elve?



Az azonos felelősségi körhöz tartozó, de egymás alternatíváját adó objektumok osztályait elkülönítjük a kód többi részétől és azokat egy közös interfészből származtatjuk.



Az alkalmazásban bevezetett interfészeket a céljuk és fajtájuk szerint külön-külön csoportokba kell szervezni.



Olyan osztályokat definiál, amelyekből legfeljebb egy példány hozható létre A származtatási hierarchiák azon osztályainak, amelyekből származtatunk, de amelyeknek nincs őosztályuk, mindig interfészeknek kell lenniük.



Két objektum sohase kerüljön közvetlen kapcsolatba, csak egy interfész közvetítésével.

10. kérdés

1 / 1 pont

Mi a látogató tervezési minta lényege?



Egy objektum reprezentációját (adattagjait) egy másik objektumban definiáljuk, amelyet majd kompozícióval csatolunk az első objektumhoz azért, hogy ez a reprezentáció rugalmasan, akár futási időben is kicserélhető legyen.



Egy objektum metódusának működése a saját osztályán kívül a neki adott paraméter-objektum osztályától függ, de ez a függőség nem jelenik meg elágazás formájában a kódban.



Egy objektum metódusának egyik paramétere egy olyan objektumra hivatkozik, amely egy közös őosztályú objektum-készletből választható ki, és ezáltal a metódus működése rugalmasan változtatható lesz.



Egy őosztálytól örökölt metódus működését az alosztályban felüldefiniált más (ún. látogató) metódusok működésétől tesszük függővé.

Kvízeredmény: **8** az összesen elérhető 10 pontból