

Eseményvezérelt és Vizuális Programozás ZH 2020. őszi – MINTA KONZULTÁCIÓRA

A ZH során egy 3 projektből álló solutiont kell létrehozni, ezekbe készítsd el az alábbi feladatok megoldását, majd a laborokhoz hasonlóan github pull request formájában add be a befejezési időpontig.

A solutionbe az alábbi típusú és nevű 3 projektet hozz létre:

- Console Application (.NET Core), C#: "ConsoleApplication" néven
- Class Library (.NET Standard) C#: "Common" néven
- xUnit Test Project (.NET Core) C#: "Tests" néven

Ne feledd, hogy egyik projektből csak akkor tudod elérni a másokban lévő dolgokat, ha a projekt References részében felveszel rá egy hivatkozást a Solution Explorerben.

Commonban interfész és default implementáció létrehozása

3p

Töröld az alapból létrejövő Class.cs-t.

Hozz létre egy XXXXXXXXXXXX nevű (publikus) interfészt, benne egyetlen metódussal XXXXXXXXXXXX néven. Paramétere XXXXXXXXXXXX, visszatérési értéke XXXXXXXXXXXX.

Hozz létre egy XXXXXXXXXXXX nevű (publikus) osztályt, mely implementálja ezt az interfészt, rendelkezik egy XXXXXXXX típusú, XXXXXXXXXXXX nevű propertyvel, és a XXXXXXXXXXXX metódusa [valamit csinál]. (Utána commit!)

Testsben triviális unit teszt létrehozása

3p

A Tests projektben töröld az alapból létrejövő UnitTest1.cs fájlt.

Hozz létre egy XXXXXXXXXXXX unit teszt osztályt, benne egy XXXXXXXXXXXX nevű teszttel, ami példányosítja a XXXXXXXXXXXX, [valamit csinál]. (Utána commit!)

(Ha a XXXXXXXXXXXX nem publikus, akkor hiába adsz a Common projektre referenciát, nem fogod tudni példányosítani. Ezen kívül minden unit teszt osztálynak is publikusnak kell lennie.)

Konzol alkalmazás elkészítése

3p

A ConsoleApplication projektben hozz létre a Program osztályban egy XXXXXXXXXXXX metódust, ami [valamit csinál].

A stringek összerakásához használj string interpolációt, az IEnumerable értékeken pedig amikor csak lehet, használj a Linq fluent API-t (pl. Min() metódus a kapott számok legkisebbikének meghatározásához).

(Ezeket a metódusokat ha a fordító nem látja, "ctrl-."-ra feljárnál a System.Linq névtér használatát, ezek ebben találhatóak.)

A Main() metódusból hívd meg a XXXXXXXXXXXX metódust, hogy a XXXXXXXXXXXX egy példány [valamit csinál].

Ellenőrzésként futtasd le a konzol alkalmazást! (Utána commit!)

[Művelet elvégzése]**6p**

A Common projektben hozz létre egy (publikus) XXXXXXXXXXXX leszármazottat XXXXXXXXXXXX néven, mely [valamit csinál]. (Utána commit!)

Unit teszt**3p**

A Tests projektben hozz létre egy (publikus) XXXXXXXXXXXX osztályt, mely egy XXXXXXXXXXXX nevű tesztben ellenőrzi, hogy XXXXXXXXXXXX.

(Ha valamiért nem működik a programod, a unit teszteket is tudod debuggolni úgy, hogy a Test Explorer ablakban a teszten jobb klikk - Debug-ot választasz.) (Utána commit!)

A konzol alkalmazás kiegészítése**6p**

A ConsoleApplication projektben a Main() metódust egészítsd ki úgy, hogy [valamit csinál]. (Utána commit!)

Megoldások leadása**1p**

A megoldást a laborfeladatokhoz hasonlóan, github pull requestként add be és a laborvezetődet rendeld hozzá reviewerként. Figyelj rá, hogy a forráskód esztétikus is legyen (felesleges üres sorok mellőzése, kikommentezett forráskód ne legyen), a commitokban csak verziókövetendő fájlok legyenek (forráskód, projekt és solution file igen, bin és obj könyvtár és .user file nem). A pull request szövegébe pedig rakj bele egy screenshotot, amin látszanak a Test Explorerben a zöld unit tesztek, valamint a konzolos alkalmazás futási eredménye. (A konzol alkalmazás ablaka ha futtatás után egyből bezárul, akkor Ctrl-F5-tel indítsd el, akkor biztosan nyitva marad.)

A ZH befejezési ideje az utolsó commit és pull request időpontjára vonatkozik. Későbbi leadásra percenként -1 pont jár.

A leadásnál a konzol és test explorer screenshot fontos, hogy szerepeljen. Ha elmarad, az workflow hiba és további pontlevonással jár. (Sorry, 118 solutiont egyesével lefuttatni csak a futás ellenőrzéséért nagyon sok idő.)

Néhány példa, hogy a metódusoknak miket kell elvégeznie, tesztelnie

- Négyzetszámok felsorolása
- Oszthatóság tesztelése
- Képlettel adott sorozat elemeinek a generálása, pl. „kovetkezo = elozo * 3 – 2”.
- Egy IEnumerable-ben valamilyen egyszerű felételnek megfelelő elemek összegének meghatározása.