



---

# Kölcsönhatás diagramok

# Célkitűzés



Olvasni tudják az alap UML kölcsönhatás diagramok (kommunikáció és szekvencia) diagramok jelöléseit.

# Bevezetés



Miért léteznek az objektumok?

Azért, hogy a rendszer valamilyen céljának megfelelően végrehajtsanak valamilyen tevékenységet.

A kölcsönhatás diagramok a rendszer dinamikájának modellezésére szolgálnak.

Hogyan változtatják meg az objektumok az állapotukat.

Hogyan lépnek egymással kölcsönhatásba (üzenet küldés/ továbbítás) ?

# Kommunikáció és szekvencia diagramok



A kölcsönhatás diagram két UML diagram típus általánosítása.

A kommunikáció diagram: Az objektumok köré szervezett kölcsönhatások valamint az egymás közötti kapcsolatok illusztrálására szolgál,

Szekvencia diagram: Az objektumok közötti kölcsönhatásokat időben elrendezve mutatja.

# Kommunikáció és szekvencia diagramok



Mindkét diagram típus szemantikusan ekvivalens, azonban nem teljesen ugyanazt az információt mutatja.

*Kommunikáció diagram* hangsúlyozza az objektumok strukturális szervezését, míg a *szekvencia diagram* az üzenetek időbeli elrendezését.

Kommunikáció diagram explicit módon mutatja az objektumok közötti kapcsolatokat, míg a szekvencia diagramban a kapcsolatokat csak implicit módon jelennek meg.

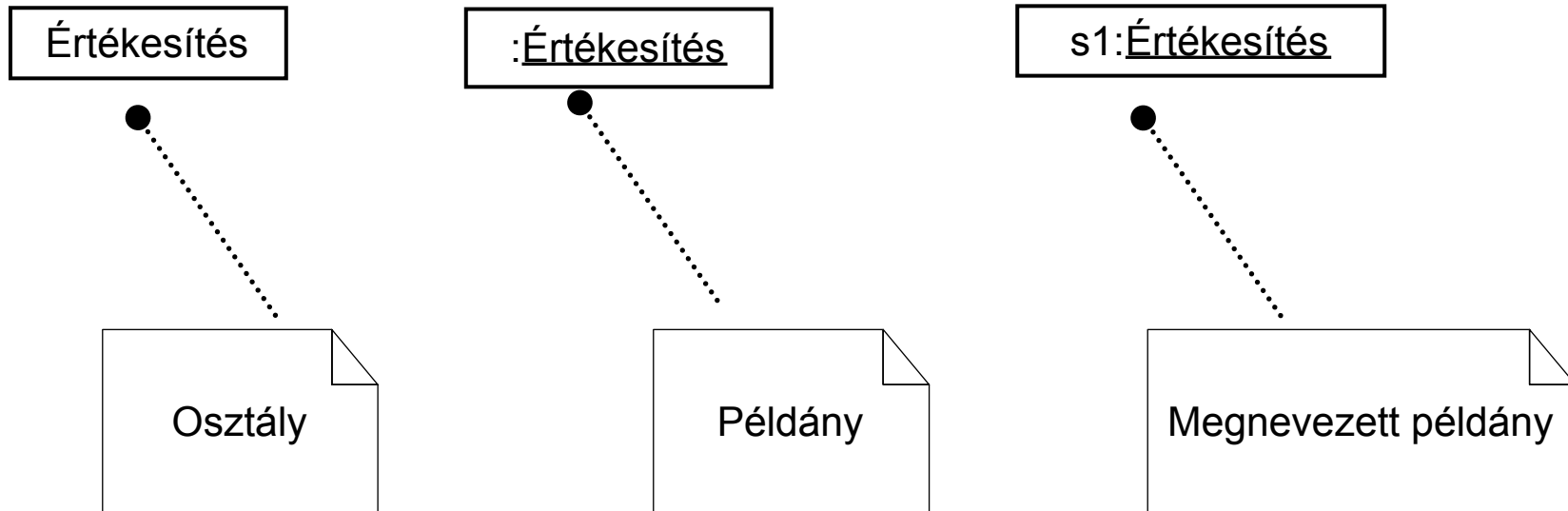
# A kölcsönhatás diagramok értékesek



A kölcsönhatás diagramok a programozók számára egy inspiráló, gondolatgazdag, kohézív, közös kiindulópontot jelent.

Mintázatok, elvek, és idiómák alkalmazhatók a kölcsönhatás diagramok minőségének javítására.

# A kölcsönhatás diagram jelölések



# Kommunikáció diagram



Az objektumokat számozott (szekvenciálisan) nyilakkal kötik össze a kapcsolatok mentén az információ áramlás érzékeltetésére.

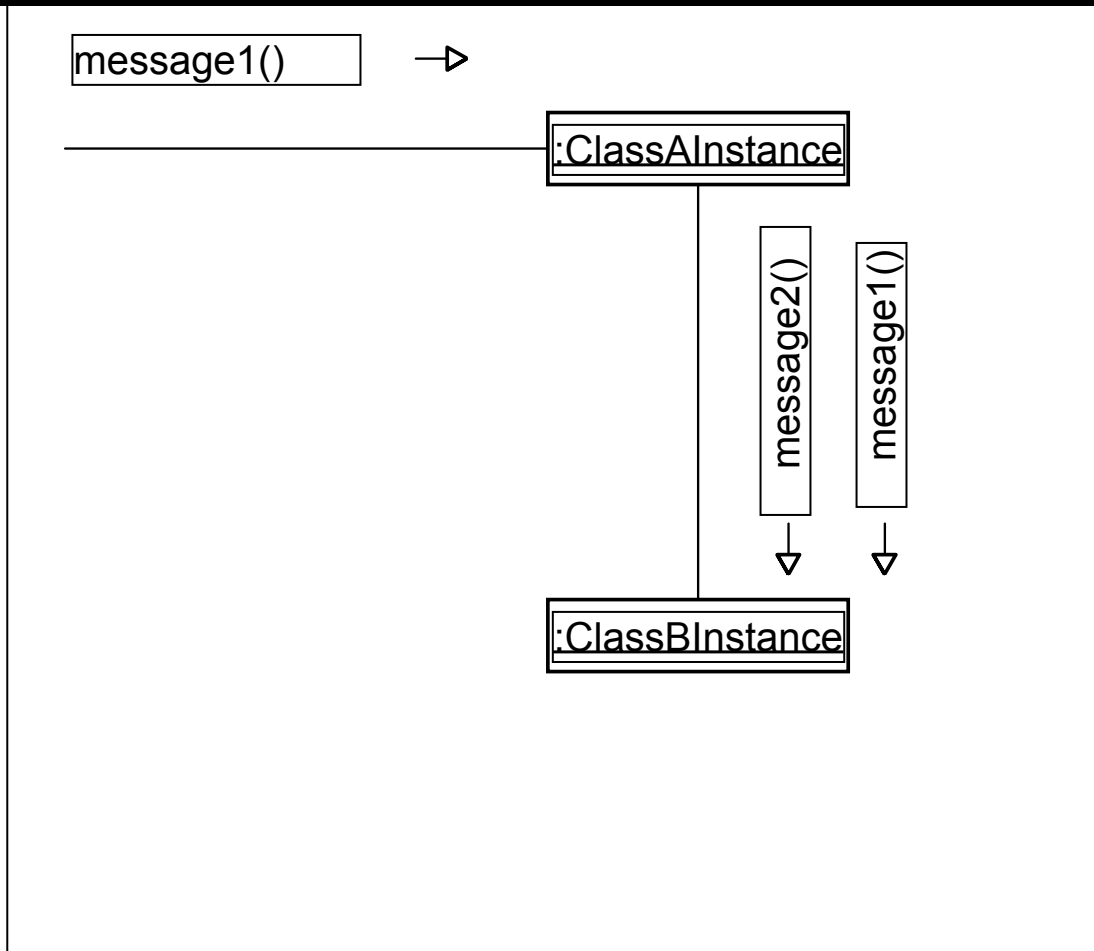
A nyilak a kölcsönhatás kezdeményezőjétől indulnak.

Az objektum, amelyre a nyíl mutat a *cél objektum*.

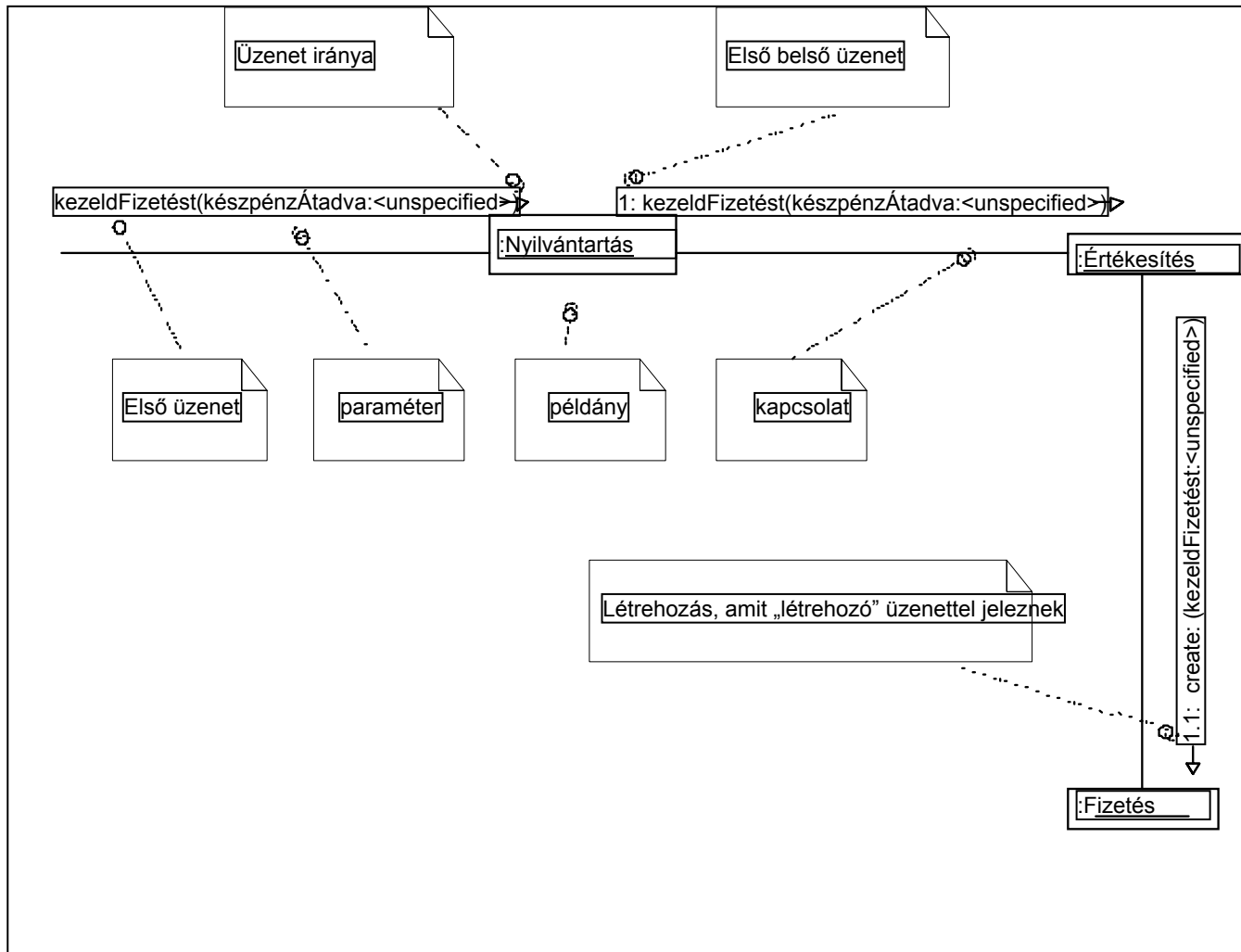
A nyilakat a továbbított üzenettel címkézik meg.



# Példa kommunikáció diagramra



# Példa kommunikáció diagramra : kezeldFizetést



# A kommunikáció diagram jelölései

---



Kapcsolat - két objektumot összekötő útvonal (egy kapcsolat példánya)

Üzenet -üzenet kifejezés jelenít meg, amelyet az objektumok közötti nyíllal ellátott vonal ábrázol

Sorszám - az információáramlások felhasználásának sorrendjét jeleníti meg.

# A kommunikáció diagram jelölései (2)



Feltételes üzenet

Sorszám [ *variable = value* ]: message()  
[ *változó = érték* ]: üzenet()

*Üzenetet* csak akkor továbbítanak, ha a kiértékelés eredménye *igaz*

Iteráció (ciklus) (Iteration (Looping))

Sor szám \* [ *i := 1..N* ]: üzenet()

Seq. Number \* [ *i := 1..N* ]: message()

"\*" kötelező; [ ... ] klóz opcionális

# Szekvencia diagram



A használati eset egy forgatókönyvének felel meg.

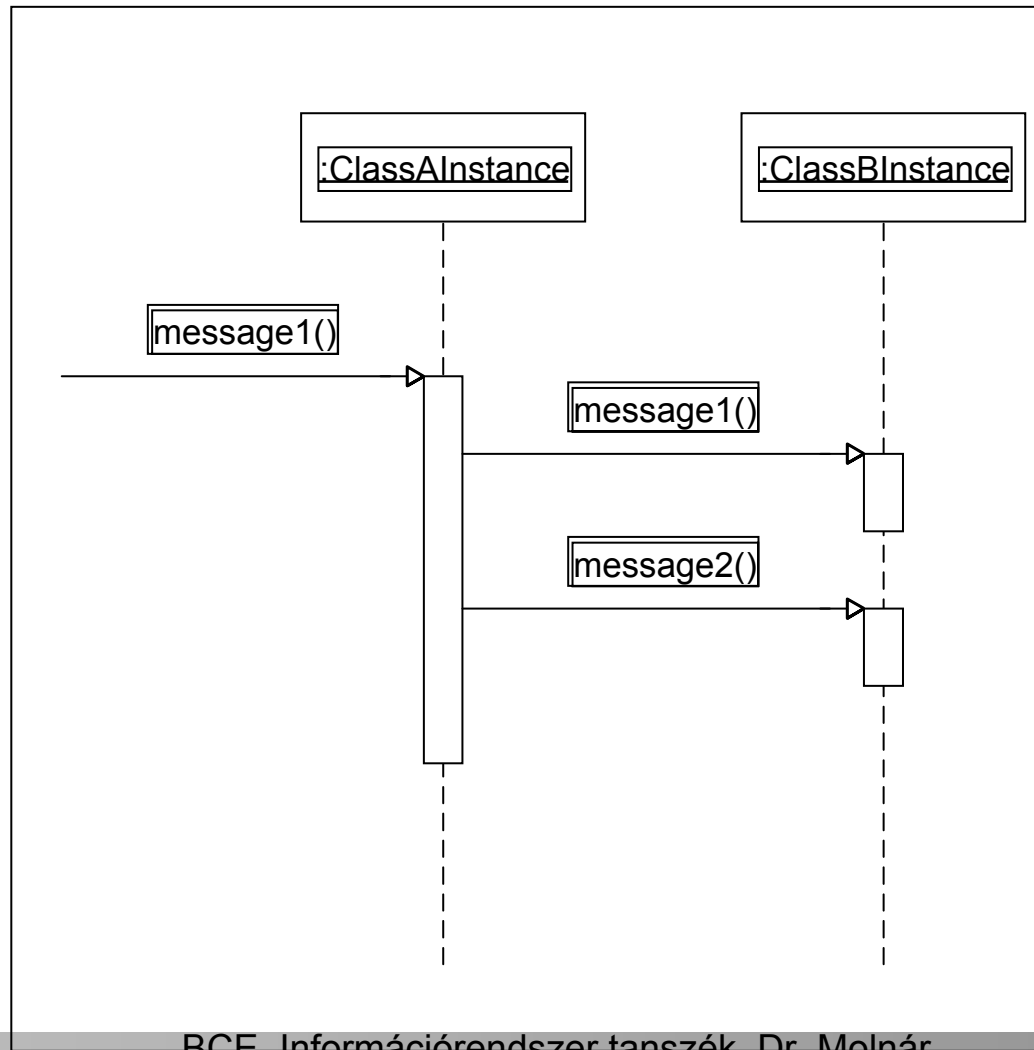
A rendszeren belül egy műveletet modellez.

Mindegyik forgatókönyvre azonosítja az objektumokat

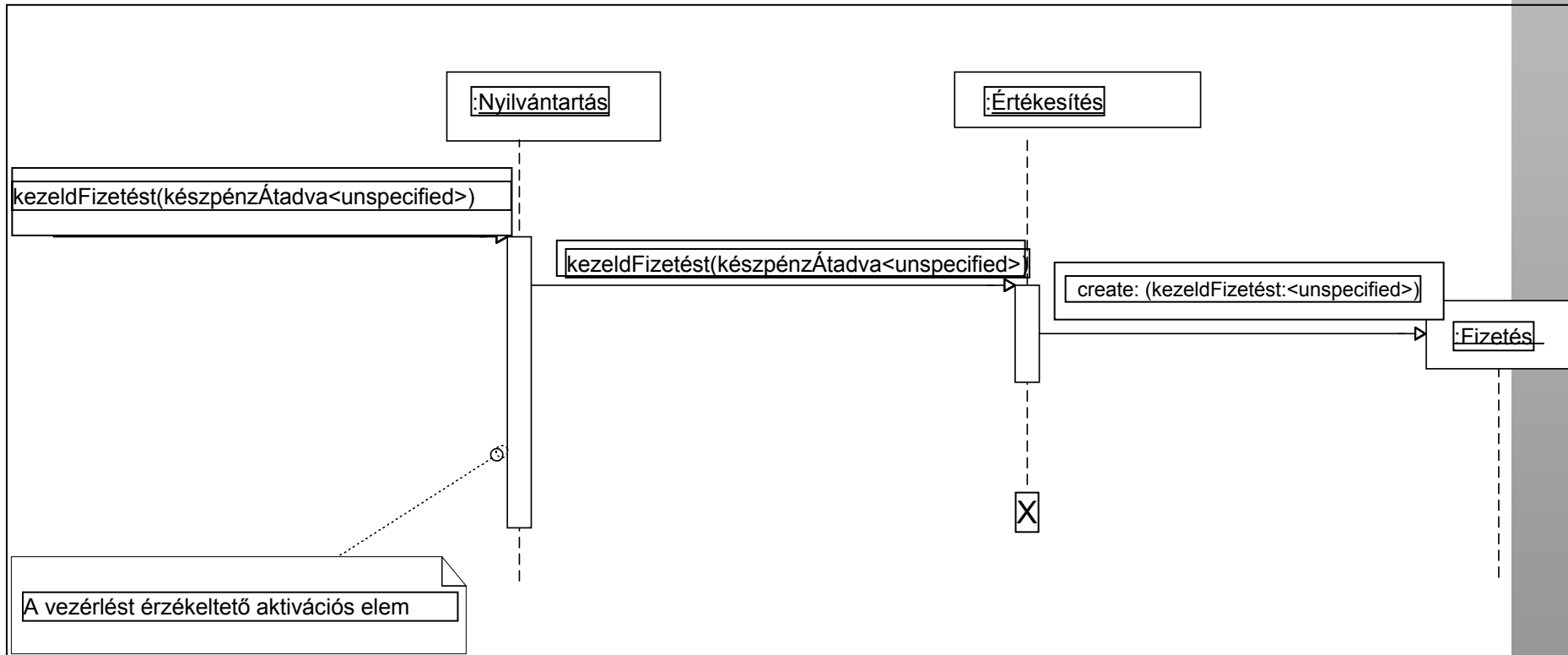
Azonosítja a továbbított üzeneteket, amelyek a forgatókönyv során történnek.

Azonosítja az egyes tevékenységekre az elvárt válaszokat, reakciót.

# Szekvencia diagram példa



# Példa szekvencia diagramra: kezelFizetést



# A szekvencia diagram alapjelölései

---



Kapcsolatok - Szekvencia diagram nem mutatja a kapcsolatokat

Üzenet - egy üzent kifejezés érzékelteti egy nyíllal ellátott objektumok közötti vonalon



# A szekvencia diagram alapjelölései (2)

---



Az objektum életvonalala - a vertikális szaggatott vonal az objektumok alatt

Az objektumoknak addig nincs életvonal, ameddig nem hozzák létre

Az objektum életének végét egy „X”-el jelöljük

Az idő fentről lefelé halad az ábrán

# A szekvencia diagram alapjelölései (3)

---



Aktiválás - az az időperiódus, ami alatt egy objektum lekezeli az üzenetet (a téglalap, a szaggatott vonal mentén).

Az aktivációs téglalapok átfedhetnek egymást azért, hogy le lehessen írni az olyan helyzeteket, amikor az objektum saját maga egy másik metódusát hívja meg.

# A szekvencia diagram alapjelölései (4)



## Feltételes üzenet

*[ variable = value ]*: message()

*[ változó = érték ]*: üzenet()

*Üzenetet csak akkor továbbítanak, ha a kiértékelés eredménye igaz*

## Iteráció (ciklus) (Iteration (Looping))

\* *[ i := 1..N ]*: üzenet()

\* *[ i := 1..N ]*: message()

"\*" kötelező; [ ... ] klóz opcionális

# A kölcsönhatás diagram erőssége



## Kommunikáció diagram

Az ábra területet gazdaságosan használja fel - könnyen lehet új objektumokkal bővíteni a két dimenziós felületen.

A bonyolult elágazásokat, iterációkat, és párhuzamos tevékenységeket jobban illusztrálja.

## Szekvencia diagram

A sorrendet (szekvenciát) és az üzenetek időbeli elrendezését világosan mutatja.

Egyszerű jelöléstechnika.

# A kölcsönhatás diagram gyengeségei

---



## Kommunikáció diagram

Nehéz látni az üzenetek sorrendjét

Bonyolult jelölés rendszer

## Szekvencia diagram

Az új objektumokat csak az ábra jobb szélén lehet elhelyezni, a horizontálisan „zabálja” a területet.

# Következtetések



Az UML-ben a kezdők gyakran túlhangsúlyozzák az osztály/objektum diagram jelentőségét. A kölcsönhatás diagramok azonban több figyelmet érdemelnek.

Nincs semmilyen szabály arra vonatkozólag, hogy mikor melyik diagramot kell használni.

Vannak olyan eszközök, amelyek a két diagram típus között automatikusan konverziót tudnak végrehajtani.