

## Tartalomjegyzék

- 1 Jelenlét, kisZH
- 2 Az el?z? gyakorlat példájának megoldása
- 3 Feladatok
  - ◆ 3.1 Egy egyszer? példa
  - ◆ 3.2 Komplex osztály
  - ◆ 3.3 Mátrix osztály folytatása
  - ◆ 3.4 NFL
  - ◆ 3.5 Páciens példa átírása

### Jelenlét, kisZH

<https://docs.google.com/spreadsheets/d/1NUkW9LDpA4-i7bvUp596fTDKrzxnWKWAFnqNAT4j5yw/edit#gid=0>

### Az el?z? gyakorlat példájának megoldása

```
class Matrix:
def const(self,n,x):
self.A = []
for i in range(n):
self.A.append([])
for j in range(n):
self.A[i].append(x)
def zeros(self,n):
self.const(n,0)
def printer(self):
for row in self.A:
print row
def set(self,i,j,x):
if len(self.A)>i:
if len(self.A[i])>j:
self.A[i][j] = x
def get(self,i,j):
if len(self.A)>i:
if len(self.A[i])>j:
return self.A[i][j]
else:
return 0
else:
return 0
matrix = Matrix()
matrix.const(6,1)
matrix.printer()
matrix.set(0,0,8)
print "\n"
matrix.printer()
print matrix.get(0,0)
print matrix.get(1,1)
```

### Feladatok

## Egy egyszer? példa

- Értelmezzük az alábbi kódokat

```
class A:
    x=8
    def __init__(self):
        self.u=6

class B(A):
    y=5
    def __init__(self):
        self.v=7
a=A()
b=B()

class Super():
    def method1(self):
        pass

class Sub(Super):
    def method1(self, param1, param2, param3):
        super(Sub, self).method1()
        pass
```

## Komplex osztály

- Az előadás anyagának megfelelően írjuk meg és próbáljuk ki a Komplex osztályt

## Mátrix osztály folytatása

- hozzunk létre \_\_init\_\_() metódust a mátrix osztályunkhoz úgy, hogy az a const() metódust hívja meg
- írjunk \_\_add\_\_() metódust
- írjunk \_\_mul\_\_() metódust
- írjunk \_\_div\_\_() metódust

## NFL

- (Táblánál)

## Páciens példa átírása

- írjuk át a 4. gyakorlat páciensekkel foglalkozó példáját objektum orientáltan.