**from** decimal **import** \*  
**from** math **import** \*  
  
**class** Razlomak(object):  
 *'''Klasa razlomak'''* **def** \_\_init\_\_(self, brojnik, nazivnik = 1):  
 **if** nazivnik == 0: **raise** Exception(**'Nazivnik ne moze biti 0'**)  
 self.\_brojnik = brojnik  
 self.\_nazivnik = nazivnik  
 **def** \_\_str\_\_(self):  
 **return '%d|%d'** % (self.\_brojnik, self.\_nazivnik)  
  
 @staticmethod  
 **def** inverz(self):  
 **return** Razlomak(self.\_nazivnik, self.\_brojnik)  
  
 @staticmethod  
 **def** stvori(broj):  
 broj\_decimala = 0  
 naz = 10  
  
 **while** (broj % 1 != 0):  
 broj \*= 10  
 broj\_decimala += 1  
 naz = pow(naz, broj\_decimala)  
  
 **return** Razlomak(broj, naz)  
  
**print**(**'\*\*\* test1 \*\*\*'**)  
r1 = Razlomak(314,100)  
r2 = Razlomak.inverz(r1)  
**print**(r1, r2)  
  
**print**(**'\*\*\* test2 \*\*\*'**)  
r1 = Razlomak.stvori(3.14)  
**print**(r1)  
r2 = Razlomak.stvori(0.006021)  
**print**(r2)  
r3 = Razlomak.stvori(-75.204)  
**print**(r3)

**2.zadatak**

**class** Stvar(object):  
 broj\_stvari = 0  
  
 **def** \_\_init\_\_(self):  
 Stvar.broj\_stvari += 1  
  
 **def** \_\_del\_\_(self):  
 Stvar.broj\_stvari -= 1  
  
**print**(**'\*\*\* test 1 \*\*\*'**)  
s1 = Stvar()  
s2 = Stvar()  
s3 = s2  
**print**(Stvar.broj\_stvari)  
**del**(s2)  
**print**(Stvar.broj\_stvari)  
**del**(s3)  
**print**(Stvar.broj\_stvari)  
**del**(s1)  
**print**(Stvar.broj\_stvari)