

Egy társasházban a közösköltségek befizetéséhez az adatokat a `koltsegek.txt` tabulátorral tagolt állományban tárolják.

Készítsen programot, amelyik a befizetendő költségekről készít számlát!

Feladatok:

Rekord szerkezet:

	Melegvíz		Hidegvíz		Villany	
	előző óráállás	utolsó óráállás	előző óráállás	utolsó óráállás	előző óráállás	utolsó óráállás
Kasler Etelka	7 786	11 631	9 350	11 055	5 771	9 606
Balogh Botond	7 523	12 354	5 250	6 801	7 394	8 814
Blazsanik Barbara	8 773	10 359	9 609	9 824	9 609	11 504
Nagy Csenge	7 423	11 132	9 036	10 061	8 072	10 364
Cseppentő Lilla	8 722	9 938	7 954	12 401	7 945	11 866
Gitta Bernát	8 582	10 418	7 377	11 278	5 076	7 562
Huttka Krisztofer	5 007	9 441	8 187	9 341	8 412	9 963
Heimpold Lilla	6 492	10 620	7 306	11 980	7 063	11 402

a számlához:

`Ft[] = {10,8,3,3800,3500}` melegvíz;hidegvíz;villany;kábelTV;közösköltség

Az adatok tárolására feldolgozására készítsen osztályt a táblázat alapján

class Lakó	
- String <code>név</code> ;	<i>név mező adattárolása</i>
- int [] <code>fogyasztás</code> = new int [3];	<i>a havi fogyasztás tárolása (utolsó óráállás – előző óráállás)</i>
Konstruktor	
+ Lakó (String sor)	<i>a konstruktor a txt állomány egy sorát kapja meg</i>
szeletelés a tmp [] -be	<i>a szokásos split()...</i>
this.név = tmp [0];	<i>a név mezőt akár itt is feltölthetjük</i>
this.fillFogyasztás (tmp);	<i>egy eljárás tölti fel a fogyasztás tömböt</i>
Metódusok	
- eljárás <code>fillFogyasztás</code> (String tmp [])	<i>a feltöltéshez az indexeket kezelni kell</i>
ciklus <code>i = 0</code> -tól; <code>i < 3</code> -ig egyesével	
<code>fogyasztás [i] = tmp[i * 2 + 2] - tmp[i * 2 + 1];</code>	
ciklus vége	
eljárás vége	
+ függvény <code>String</code> <code>getNév</code> ()	<i>a név mező itt érhető el kívülről</i>
return <code>Név</code> ;	
függvény vége	
+ függvény <code>int</code> [] <code>getFogyasztás</code> ()	<i>a fogyasztás tömb így érhető el kívülről</i>
return <code>fogyasztás</code> ;	
függvény vége	
+ függvény <code>String</code> <code>getSzámla</code> ()	<i>saját maga készíti el a számláját és a főprogram kiírja a Saját Név.txt fájlba</i>
<i>//az adatokat a minta szerint formázva</i>	
String <code>fmelegviz</code> = "melegvíz: " + <code>fogyasztás[0]</code> * 10;	
String <code>fhidegviz</code> = "hidegvíz: " + <code>fogyasztás[1]</code> * 8;	
...	
<code>szumma</code> = <code>fogyasztás[0]</code> * 10 + <code>fogyasztás[1]</code> * 8 + <code>fogyasztás[2]</code> * 3 + 3800 + 3500;	
String <code>s</code> = <code>fmelegviz</code> + "\n" + <code>fhidegviz</code> + "\n" + ...+ <code>fÖsszes</code> + "\n";	
return <code>s</code> ;	
függvény vége	

Főprogram

- az adatok tárolása a **házlakói lista** adatszerkezetben
- a `koltsegek.txt` adatállomány beolvasása előltesztelő ciklussal történik és minden sort a **Lakó** osztály konstruktorának kell átadni.

A beolvasás végén az adatszerkezet:

házlakói<>

Kasler Etelka		
3845	1705	3835

Balogh Botond		
4831	1551	1420

Blazsanik Barbara		
1586	215	1895

Nagy Csenge		
3709	1025	2292

Cseppentő Lilla		
1216	4447	3921

Gitta Bernát		
1836	3901	2486

Huttka Krisztofer		
4434	1154	1551

Heimpold Lilla		
4128	4674	4339

Hozzáférés pl. a

villanyfogyasztáshoz:

`házlakói.get(i).getFogyasztás().[2]`

névhez:

`házlakói.get(i).getNév()`

Készítsen számlákat a névsorba rendezett listából a havi befizetésről állományokba is.

Az állomány neve a tulajdonos neve legyen txt kiterjesztéssel.

melegvíz:	3845 m3	38450 Ft
hidegvíz:	1705 m3	13640 Ft
villany:	3835 kWh	11505 Ft
kábel TV:		3800 Ft
közösköltség:		3500 Ft

összesen: 70895 Ft

```

eljárás adatokBe(String f_neve)
    f nyitása(f_neve, "r");
    sor = f.sorolvasása();
    ciklus amíg (sor != null)
        házlakói.add(new Lakó(sor));
        sor = f.sorolvasása();
    ciklus vége
    f.zárása ();
eljárás vége

```

előolvasás az **előltesztelő** ciklusba való belépéshez

a **Lakó** osztály konstruktorának hívása

```

eljárás rendezés()
    ...
    ha (házlakói.get(j).getNév().compareTo(házlakói.get(j).getNév()) < 0)
    ...
eljárás vége

```

rendezés névsorba feltétele

```

eljárás adatokKi(String f_neve, String név)
    Ha a fájl létezik, akkor törlés
    f nyitása(f_neve, "rw");
    f.írása();
    f.zárása ();
eljárás vége

```

érdemes törölni a fájlt, mert hiba esetén maradék sorok lehetnek benne...

```

eljárás számla()
    N = házlakói.mérete();
    ciklus i = 0-tól N-ig egyesével
        név = házlakói.get(i).getNév();
        s = házlakói.get(i).getSzámla();
        Ki: név, s;
        adatokKi(név+".txt", s);
    ciklus vége
eljárás vége

```

metódus hívások