



Java



Egy kis ismétlés nem árt

A Java nyelvben az adattípusoknak két fajtája van:

- **primitív**
- **referencia** vagy **összetett** típus

Primitív típus

Egész

Valós

Karakter

Logikai

Referencia típus

Tömb

Osztály

Interfész

Egy kis ismétlés nem árt

A primitív adattípusok egy egyszerű értéket képesek tárolni: számot, karaktert vagy logikai értéket.

- egészek
byte, short, int, long
- valósak
float, double
- egyéb
char, boolean

A String osztály

A **referencia** - összetett, komplexebb- típusok, minden esetben objektumok.

Ezek egyike a karakterlánc, más nevén String.

A Java három osztályt biztosít , melyekkel tárolhatunk, illetve manipulálhatunk sztringeket,

String

StringBuffer

StringBuilder

A String osztály

String osztályban olyan sztringeket tárolunk, amelyek értéke nem fog változni.

A **StringBuffer** osztályt akkor alkalmazzuk, ha a szövegen szeretnénk módosítani.

A **StringBuilder** osztályt - a JDK 5.0-tól - gyorsabb, mint a **StringBuffer**.

Ha a szöveg nem fog változni, használjuk a **Stringet**.

String osztály

Készíthetjük a sztringet egy *sztring konstansból*, azaz karaktersorozatból.

```
String s = "Agg atyák gatyája";
```

Vagy mint minden más objektumot a new operátorral hozhatjuk létre.

```
String s = new String ("Agg atyák gatyája");
```

A String indexelése, metódusai

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
A	g	g		a	t	y	á	k		g	a	t	y	á	j	a

A String osztály egy-két metódusa

```
int N = s.length();    // szöveg hossza
```

```
char c = s.charAt(0);  // adott indexű karakter
```

```
String kicsi = s.toLowerCase();
```

```
String nagy = s.toUpperCase();
```

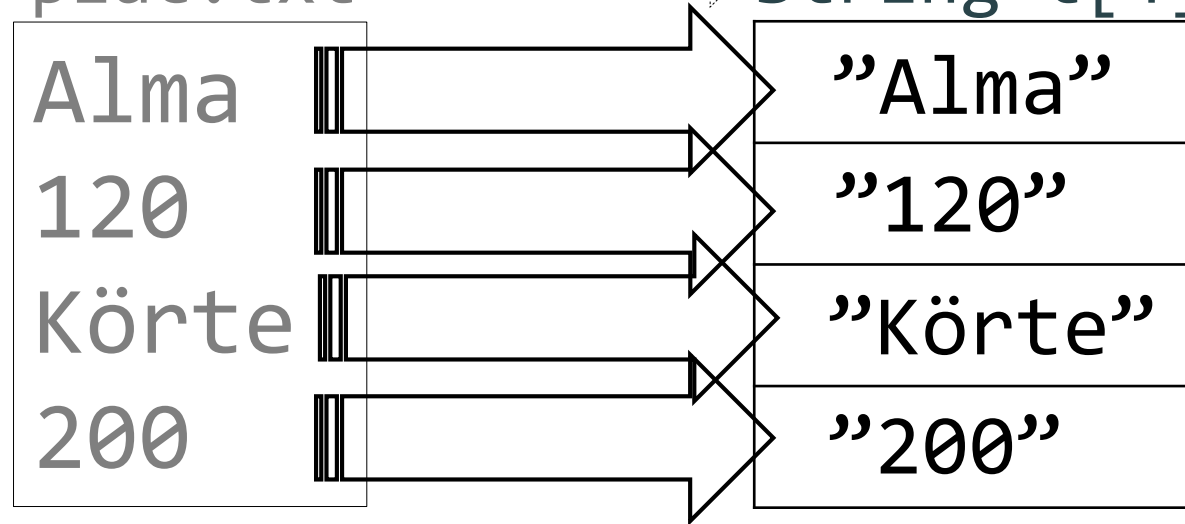
```
String s2 = s.replace(a, e); // karakterek cseréje
```

```
String s2 = s.substring(4,7); // [rész) string
```

Újra a fájlokról

A beolvasás soronként történik egy String tömbbe.

`t[i] = readLine();` `piac.txt`  `String t[4]`



A fájl sorait minden esetben mint karakterláncokat olvassuk be.

Integer osztály

A számokat, át/vissza kell alakítanunk szám típusná, hogy számolni tudjunk velük.

```
pl.: int x = Integer.parseInt(t[3])*10;
```

Az **Integer** nem véletlenül nagy kezdőbetűs,
az **Integer osztályt** jelöli, ezért van metódusa.

Beolvasási problémák

A beolvasás soronként történik egy Stringbe.

```
String s = readLine(); piac.txt —————> String t[4]
```

piac.txt

Alma,120,Körte,200

String s =

"Alma,120,Körte,200"

```
t=s.split(",");
```

"Alma,120,Körte,200"

"Alma"

"120"

"Körte"

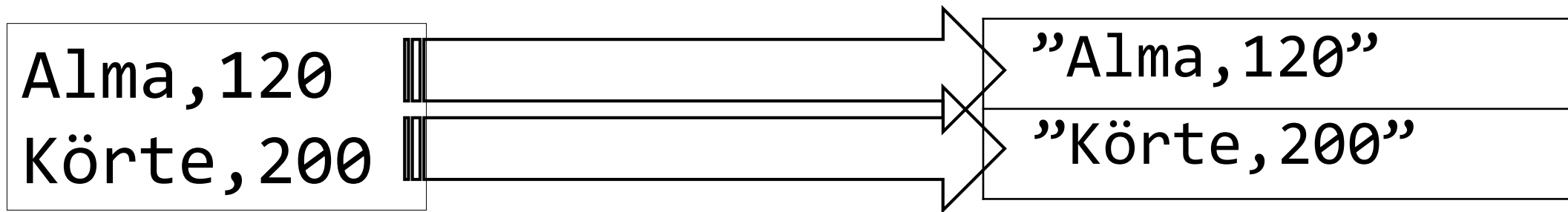
"200"

A fájl sorait minden esetben mint karakterláncokat olvassuk be.

Beolvasási problémák

A beolvasás soronként történik egy String tömbbe.

`t[i] = readLine();` `piac.txt`  `String t[4]`



A fájl sorait minden esetben mint karakterláncokat olvassuk be.