



PERSONAL
DEVELOPMENT
PROCESS

stay weird, stay different

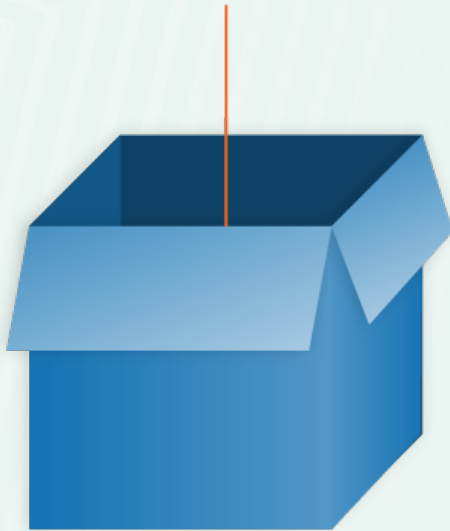
Javada ma'lumot turlari va o'zgaruvchilar.

Javada o'zgaruvchilar (variables)

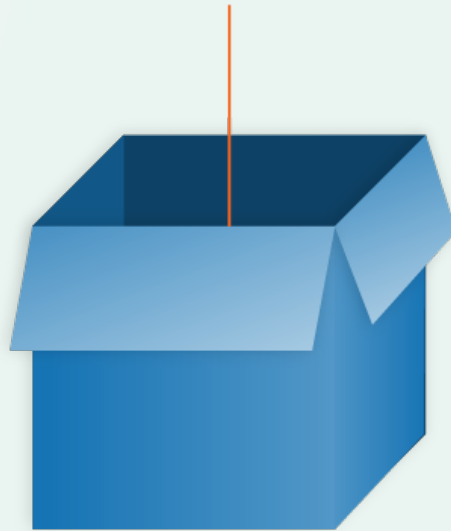
- O'zgaruvchi — ma'lum bir toifadagi ma'lumotlarni (qayta) yozish va o'qish mumkin bo'lgan xotiraning nomlangan qismi.
- O'zgaruvchini e'lon qilayotganda (yaratayotganda) uning toifasi va nomi ko'rsatiladi. Keyinchalik ularni ozgartirib bo'lmaydi.

Variables and Data Types

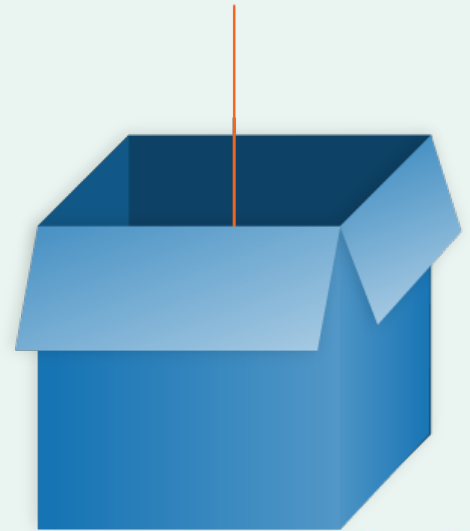
"Bob"



true



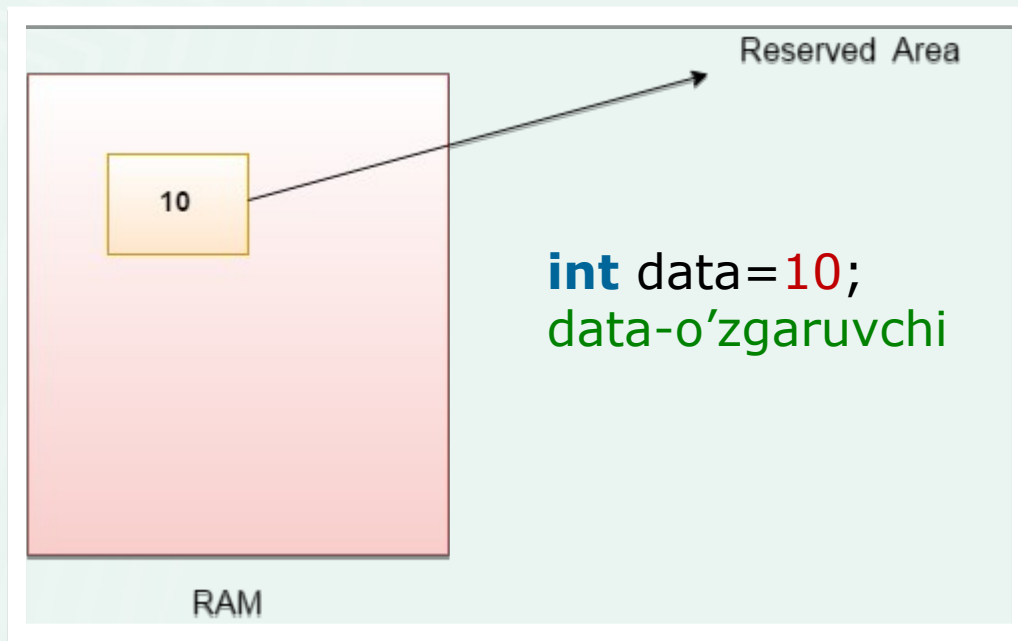
35



How to Declare a Variable In Java

```
data_type variable_name = value;
```

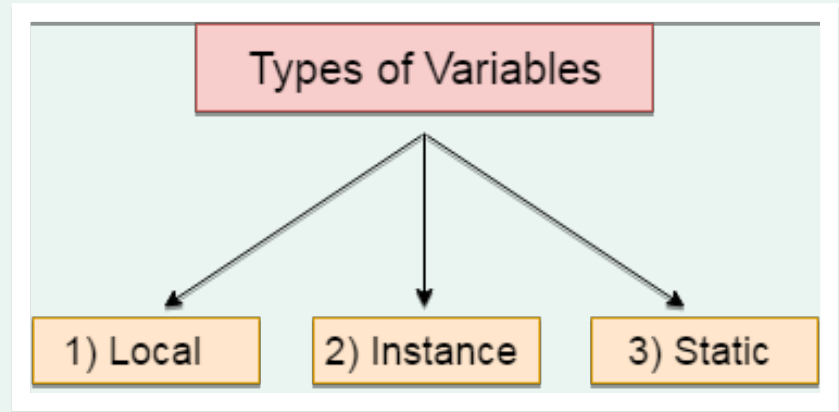
Variables



O'zgaruvchi turlari

Javada 3 xil turdagi
o'zgaruvchi mavjud:

- ❖ Local
- ❖ instance
- ❖ static



1) Local Variable

Method ichida e'lon qilingan o'zgaruvchilarga local o'zgaruvchi deyiladi.

2) Instance Variable

Klassni ichida va methoddan tashqarida e'lon qilingan o'zgaruvchilarga instance o'zgaruvchi deyiladi.

3) Static variable

Klassni ichida va static deb e'lon qilingan o'zgaruvchilarni static o'zgaruvchi deyiladi.

U local bo'lishi mumkin emas.



```
class A{  
    int data=50; //instance variable  
    static int m=100; //static variable  
    void method(byte b ){ //local variable  
        int n=90;    //local variable  
    }  
}
```


Javada o'zgaruvchilarni nomlash qoidalar

❖1. O'zgaruvchi nomi (kichik) harflardan (Unicode) boshlanishi kerak hamda raqam va «_» belgilardan tashkil topgan bo'lishi lozim. Texnik jihatdan o'zgaruvchi nomi «\$» belgi bilan boshlanishi mumkin, lekin Java Code Conventions (<https://www.oracle.com/technetwork/java/codeconventions-150003.pdf> 12.09.1997) bo'yicha bu taqiqlangan. Bundan tashqari ushbu kelishuv bo'yicha bu belgi umuman hech qayerda foydalanilmaydi. Kelishuv bo'yicha o'zgaruvchi nomi aynan kichik harflardan boshlanishi kerak (katta harflar bilan klasslar nomlanadi). O'zgaruvchi nomida probel ishlatilishiga ruxsat etilmaydi.



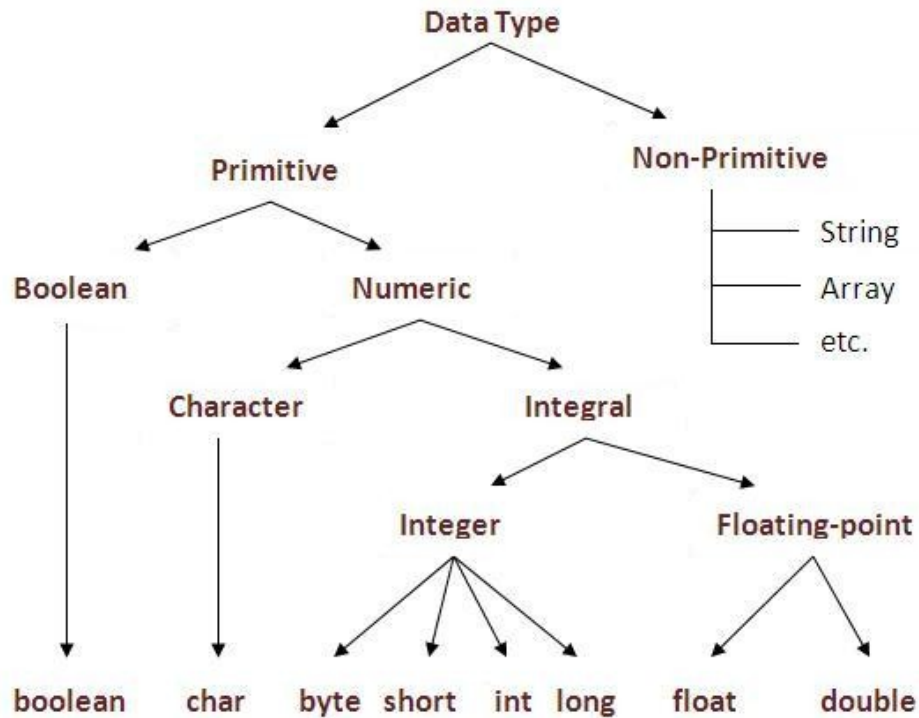
- ❖ O'zgaruvchi nomi java kalit so'zi bo'lmasligi kerak.
- ❖ O'zgaruvchi nomidagi katta va kichik harflar farqlanadi. `newVariable` va `newvariable`
- ❖ O'zgaruvchi nomini tanlaganda sirli abbreviaturalar o'rniga to'liq so'zdan foydalanish maqsadga muvofiq. Bu esa sizning kodingiz o'qish va tushunish uchun qulay qiladi.
- ❖ Agar o'zgaruvchiga tanlangan nom bitta so'zdan iborat bo'lsa kichik harflar bilan yoziladi. Agarda bittadan ko'p so'zdan iborat bo'lsa navbatdagi har bir so'z kata harf bilan ajariladi (camelCase).
- ❖ Agar o'zgaruvchi o'zgarmas(doimiy) qiymatni saqlasa har bir so'z katta harflarda yoziladi va so'zlar “_” belgisi bilan ajratiladi. Masalan:
- ❖ `static final int NUMBER_OF_HOURS_IN_A_DAY=24;`

Data Types in Java

Ma'lumot turi deganda o'zgaruvchilarda saqlash mumkin bo'lgan turli qiymatlar tushuniladi. Javada ular 2 toifaga ajratiladi:

- Primitive data types
- Non-primitive data types

Javada ma'lumot turlari



boolean

- boolean toifasi 1 bit ma'lumotni ifodalash uchun ishlatiladi.
- Faqat ikkita qiymat qabul qilishi mumkin : true va false
- Default qiymati false
- Example: boolean one = true;

Sanoq sistemalari

Eng ko'p tarqalgan 10 lik sanoq sistemasidir. Lekin u yagona emas. O'tmishda ko'pchilik xalqlar turli sanoq sistemalaridan foydalanishgan. Hozirgi kunda ularning deyarli barchasi 10 lik sistemaga o'tishgan. O'nlik sanoq sistemasida 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 raqamlari ishlatiladi. Ularni birlashtirgan holda yangi son hosil qilish mumkin.

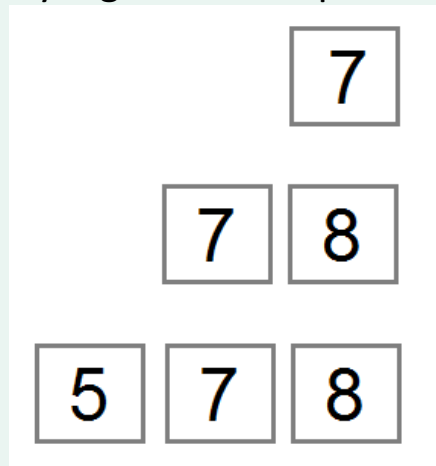
Agarda sizda:

-1 ta yacheyka bo'sa 0 dan 9 gacha

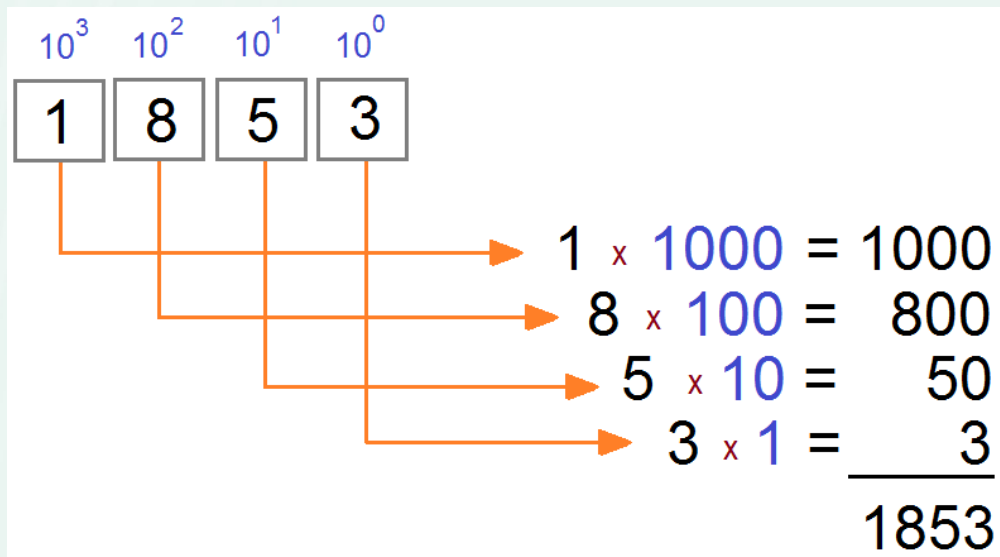
-2 ta yacheka bo'lsa 0 dan 99 gacha

-3 ta yacheka bo'lsa 0 dan 999 gacha

bo'lgan sonni ushbu yacheka(lar)ga yozishingiz
mumkin.



Yacheykalar o'ngdan chapga qarab ketma-ketlikda koeffisientga ega 10^0 , 10^1 , 10^2 ...



BIT

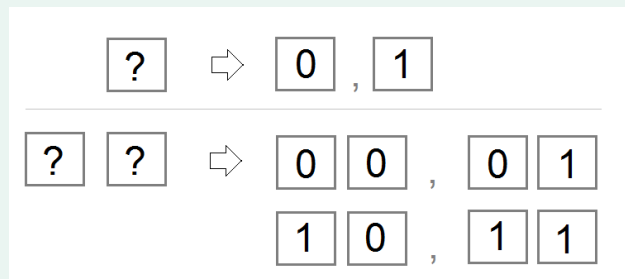
BIT -bu Binary digIT so'zining qisqartirilgan shakli (axborot o'lchov birligi). Bir **bit** 0 yoki 1 qiymatga ega va u komputerdagi eng kichik birlikdir. 0, 1 – bu ikkilik sanoq sistemasining asosiy raqamlari

Xuddi o'nlik sanoq sistemasidagi kabi siz:

-1 ta yacheyka bo'sa 0 va 1 ni,

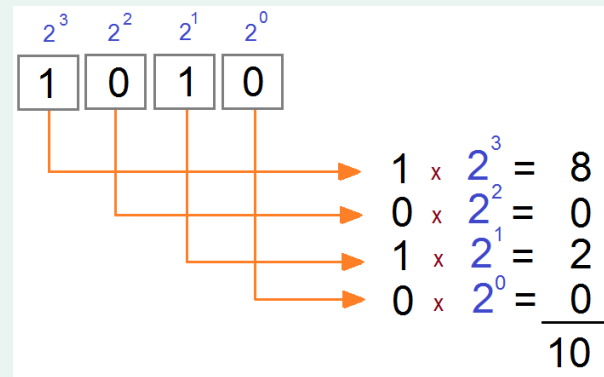
-2 ta yacheka bo'lsa **00, 01, 10, va 11 ni**,

ushbu yacheka(lar)ga yozishingiz mumkin.



Xuddi o'nlik sanoq sistemasidagi kabi
yacheykalar o'ngdan chapga qarab
ketma-ketlikda koefitsientga ega.

Demak, agarda sizda 2 lik sanot sistemasida:
-2 ta yacheyka bo'lsa unga eng kata son 11(2) ni
(onlikda 3 ga teng)
-3 ta yacheyka bo'lsa unga eng katta son 111(2)
ni (onlikda 7 ga teng) yozishingiz mumkin.



Box Numbers	Maximum Number (Base-2)	Convert to Base-10
1	1	$1 (2^1 - 1)$
2	11	$3 (2^2 - 1)$
3	111	$7 (2^3 - 1)$
4	1111	$15 (2^4 - 1)$
5	11111	$31 (2^5 - 1)$
6	111111	$63 (2^6 - 1)$
7	1111111	$127 (2^7 - 1)$
8	11111111	$255 (2^8 - 1)$
9	111111111	$511 (2^9 - 1)$



Nima uchun computer o'nlik emas ikkilik sanoq tizimida ishlaydi?

Komputer millionlab tranzistorlardan (yoqib-o'chirgich, electronic switches) foydalangan holda ishlaydi. Har bir transistor ikkita holatga ega: **yoqilgan** yoki **o'chirilgan**. Yoqib-o'chirish holati binar (ikkilik) ma'lumotni ifodalashi mumkin, ha yoki yo'q, to'g'ri yoki noto'g'ri, 1 yoki 0. Kompyuterda axborotning asosiy o'lchov birligi ikkilik belgidir (binary digit). Kompyuter o'zida juda ko'p turli xil ma'lumotlarni ifodalay olishi mumkin bo'lishiga qaramasdan barcha ifodalar tranzistorning ON/OFF holatiga keltiriladi (mos holda 1 va 0).



byte

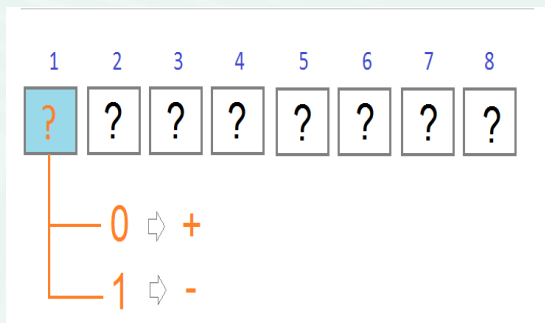
- Byte toifasi hajmi 8-bit bo'lgan butun sonlarni ifodalash uchun qo'llaniladi.
- Minimum qiymati -128 (-2^7)
- Maximum qiymati 127 (inclusive) ($2^7 - 1$)
- Default qiymati 0
- Example: byte $a = 100$, $b = -50$;

byte ning qiymatlar diapazoni $\rightarrow [-128, 127]$

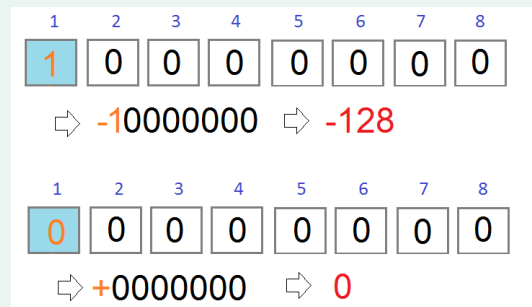
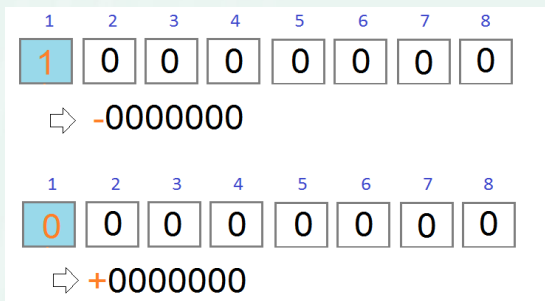
Nima uchun $[-128, 128]$ emas?



Chunki birinchi bit ishorani bildiradi.



Birinchi bit ishora va qolgan 7 ta bit 0..127 bo'lgan son. Demak byte diapazoni [-127,127] bo'lishi kerak. Lekin:



short

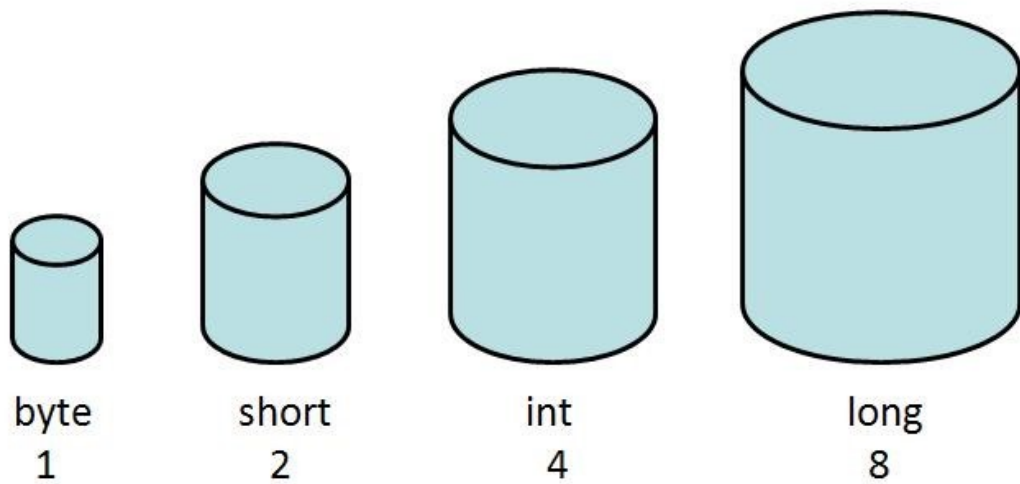
- Short toifasi hajmi 16-bit bo'lgan butun sonlarni ifodalash uchun qo'llaniladi.
- Minimum qiymati $-32,768$ (-2^{15})
- Maximum qiymati $32,767$ (inclusive) ($2^{15} - 1$)
- Default qiymati 0.
- Example: `short s = 10000, r = -20000, x; //x=0`

int

- int toifasi hajmi 32-bit bo'lgan butun sonlarni ifodalash uchun qo'llaniladi.
- Minimum qiymati - 2,147,483,648 (-2^{31})
- Maximum qiymati 2,147,483,647(inclusive) ($2^{31} - 1$)
- int barcha butun sonli toifalarning default toifasi hisoblanadi.
- Default qiymati 0
- Example: `int a = 100000, b = -200000;`

long

- Long toifasi hajmi 64-bit bo'lgan butun sonlarni ifodalash uchun qo'llaniladi.
- Minimum qiymati $-9,223,372,036,854,775,808(-2^{63})$
- Maximum qiymati $9,223,372,036,854,775,807$ (inclusive)($2^{63} - 1$)
- Default qiymati 0L
- Example: long a = 100000L, b = -200000L, x=9;



float

- float toifasi 32- bitli vergulli sonlarni ifodalash uchun ishlatiladi.
- Default qiymati 0.0f
- Example: float f1 = 234.5f



double

- double toifasi 64- bitli vergulli sonlarni ifodalash uchun ishlatiladi.
- Ushbu toifa vergulli sonlar uchun default toifa hisoblanadi.
- Default value is 0.0d
- Example: double d1 = 123.4



char

- char toifasi 1 dona 16-bitli (2byte) Unicode belgini ifodalash uchun ishlatiladi.(Unicode jadvali)
- Minimum qiymati '\u0000' (or 0)
- Maximum qiymati '\uffff' (or 65,535 inclusive)
- char toifasi yordamida istalgan belgini ifodalash mumkin.
- Example: char letterA = 'A', number='1';