

به نام خدا

داده‌های اصلی

در جاوا

شکیلا طایفه

دانشگاه الزهرا

آبان ماه ۱۳۹۷

## داده‌های اصلی

در زبان جاوا ۸ نوع داده اصلی وجود دارد. انواع داده اصلی در زبان جاوا از قبل تعریف شده اند و قابل گسترش نیستند. به عبارت دیگر آن‌ها به عنوان پایه انواع داده‌های دیگر هستند. انواع داده اصلی دارای نام‌هایی هستند که این نام‌ها جزء کلمات کلیدی است. در ادامه به معرفی این انواع داده می‌پردازیم و آن‌ها را شرح خواهیم داد.

### byte

نوع داده byte دارای خواص زیر است:

این نوع داده ۸ بیت حافظه لازم دارد.

فقط می‌توان عدد بدون ممیز در این نوع داده ذخیره کرد.

این نوع داده کوچکترین عددی را که می‌تواند در خود نگه دارد ۱۲۸- است.

بزرگترین عددی که این نوع داده می‌تواند در خود نگه دارد ۱۲۷ می‌باشد.

مقدار پیش فرض این نوع داده ۰ است، به این معنی که وقتی یک متغیر از نوع byte تعریف می‌کنیم کامپایلر مقدار آن را ۰ قرار می‌دهد.

اگر اعداد شما در برنامه‌نویسی زیاد بزرگ نیست و از ممیز هم استفاده نمی‌کنید می‌توانید از این نوع داده استفاده کنید زیرا که در حافظه صرفه جویی می‌کند و یک چهارم نوع داده int فضا لازم دارد. کد زیر مثالی از تعریف متغیر از نوع داده byte است.

```
byte a=100;  
byte b=-43;
```

### short

این نوع داده اعداد بدون ممیز را در خود نگهداری می‌نماید.

نوع داده short یک نوع داده علامت‌دار ۱۶ بیتی است.

کوچکترین مقداری که این نوع داده می‌تواند در خود نگهداری نماید ۳۲۷۶۸- می‌باشد.

بزرگترین مقداری که این نوع داده می‌تواند در خود نگهداری نماید ۳۲۷۶۸ می‌باشد.

مقدار پیش فرض این نوع داده ۰ است.

حافظه مصرفی این نوع داده نصف نوع داده int است.

```
short s=10000;  
short r=-3400;
```

### int

نوع داده int یک نوع داده عددی علامت‌دار ۳۲ بیتی است.

کوچکترین عددی که این نوع داده می‌تواند در خود نگهداری نماید ۲۱۴۷۴۸۳۶۴۸- می‌باشد.

بزرگترین عددی که این نوع داده می‌تواند در خود نگهداری کند ۲۱۴۷۴۸۳۶۴۷ می‌باشد.

عموماً برای متغیرهای اعداد صحیح از این نوع داده استفاده می‌شود ولی برای استفاده بهینه‌تر از حافظه می‌توان از انواع داده قبلی هم استفاده کرد. ولی امروزه به علت بالا بودن حجم حافظه‌ها صرفه‌جویی در حافظه مسئله حادی نیست. مقدار پیش فرض این نوع داده ۰ است.

```
int m=60;  
int c=-200000;  
int h=800000;
```

## long

نوع داده long، یک نوع داده برای نگهداری اعداد صحیح علامت‌دار است که ۶۴ بیت حافظه اشغال می‌کند. کوچکترین عددی که این نوع داده می‌تواند در خود نگهداری کند  $-9223372036854775808$  است. بزرگترین عددی که این نوع داده می‌تواند در خود نگهداری کند عدد  $9223372036854775807$  می‌باشد. این نوع داده زمانی استفاده می‌شود که بخواهیم با اعداد صحیح بسیار بزرگ کار کنیم. مقدار پیش‌فرض این نوع داده ۰L است.

```
long a=1000000L;
```

## float

این نوع داده برای نگهداری اعداد ممیزدار با ۳۲ بیت است. این نوع داده بر اساس استاندارد ممیز شناور IEEE ۷۵۴ کار می‌کند. ( مؤسسه مهندسان برق و الکترونیک که به IEEE (The Institute of Electrical and Electronics Engineers) معروف است، یک سازمان بین‌المللی حرفه‌ای و ناسودبر است. خواست این مؤسسه کمک به پیش‌برد فناوری به طور گسترده و حوزه‌های وابسته به مهندسی برق و کامپیوتر و همچنین زمینه‌های وابسته به طور خاص است. اصطلاح ممیز شناور به این واقعیت اشاره دارد که ممیز (ممیز اعشاری یا ممیز دودویی که در رایانه‌ها رایج است) می‌تواند «شناور» باشد، یعنی می‌تواند هرجایی بین ارقام معنی‌دار یک عدد، قرار بگیرد. مکان ممیز به صورت جداگانه در نمایش درونی مشخص می‌شود و می‌توان ممیز را درک رایانه از نماد علمی دانست.) مقدار پیش فرض این نوع داده ۰.۰f است. این نوع داده برای جاهایی که دقت بسیار مهم است مورد استفاده قرار نمی‌گیرد. مثلاً برای نگهداشتن واحدهای پولی از این نوع داده استفاده نمی‌شود چون دقیق نیست.

```
float f=234.2f;
```

## double

این نوع داده یک نوع داده ۶۴ بیتی دقیق است. این نوع داده از استاندارد ممیز شناور IEEE ۷۵۴ استفاده می‌کند.

این نوع داده ، نوع داده پیش فرض برای اعداد اعشاری می باشد.  
این نوع داده با این که از float دقیق تر است اما همچنان خطا دارد و نباید برای مقادیر پولی مورد استفاده قرار گیرد.  
مقدار پیش فرض این نوع داده d ۰.۰ می باشد.

```
double d=123.4;
```

## boolean

این نوع داده فقط دو مقدار true,false را در بر می گیرد.  
این نوع فقط برای نگهداری نتایج عبارات شرطی مورد استفاده قرار می گیرد.  
مقدار پیش فرض این نوع داده false است.

```
boolean b=true;
```

## char

این نوع داده برای نگهداری کاراکترهای یونیکد می باشد.  
این نوع داده ۱۶ بیت فضا از حافظه برای خود می گیرد.

```
char letter='a';
```