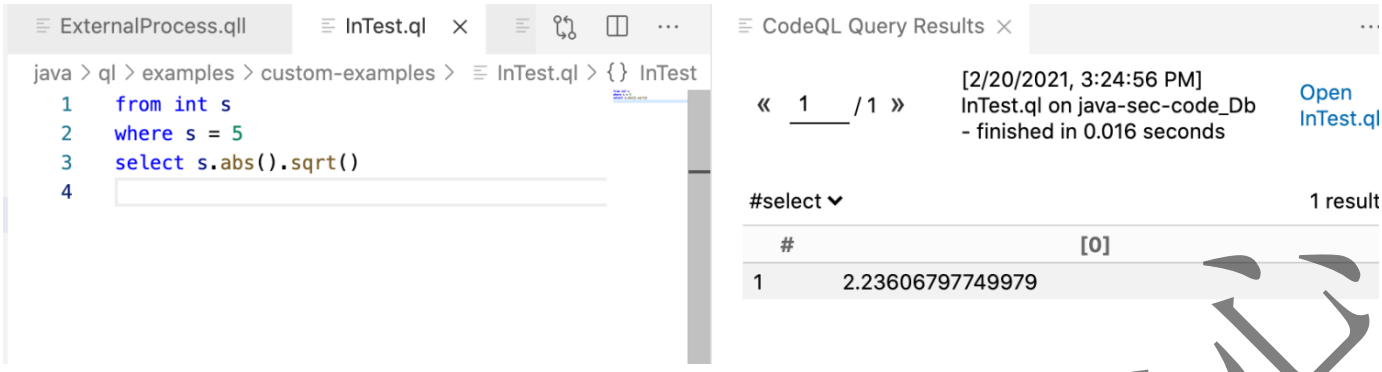
QL语言基本数据类型

引言

现在，我们开始学习QL语言中的基本数据类型，包括整型、浮点型、日期型、布尔型以及字符串类型。需要注意的是，对于QL语言来说，其支持的数据类型都带有相应的内建函数——通俗来说，就是系统已经为我们写好的函数，我们直接拿来就能用了。举例来说，如果我们想求一个整数的绝对值，直接调用内建函数abs()即可，例如-6.abs()。更一般地说，调用某种类型的变量的通用形式为：直接在变量后面加上一个点号，然后加上要调用的内建函数即可。同时，我们还可以通过点号将多个函数串联起来，也就是对变量连续进行多种处理，例如，对于一个整型变量a，先求绝对值，再开平方，我们可以将这个处理过程表示为：a.abs().sqrt()。

int类型的数据包含的内建函数

```
/** 通过QL计算int型整数5的绝对值开平方 **/
from int i
where i = 5
select s.abs().sqrt()
```

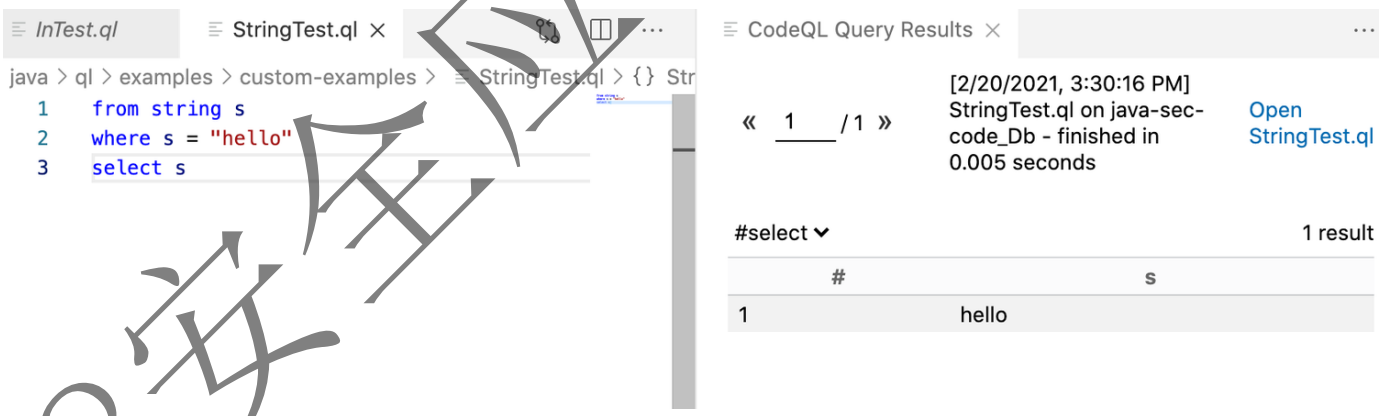


字符串类型

字符串类型的变量用来存放以双引号开头和结尾的字符序列，即字符串。例如：

```
from string s
where s = "hello"
select s
```

其中，在from语句中，我们定义了一个字符串类型的变量s，然后，我们在where语句中，将字符串”hello”赋值给了变量s，最后，我们在select语句中返回变量s的值。如果在查询控制台运行上述代码的话，运行结果将为：



注意，上面的运行结果中，并没有出现双引号。这是因为，双引号是一个特殊字符：字符串通常使用双引号"..."来表示开始和结束，所以双引号本身不会显示出来。读到这里，读者可能会问：如果字符串本身恰

好包含一个"字符的话，那该怎么表示呢？这个时候，就该转义字符\上场了。具体来说，只要在双引号前面加上一个反斜杠，双引号就不再表示字符串的开始或结束位置的指示符，而是表示双引号自身了，具体如下所示：

```
from string s, string b
where s = "hello" and b= "he\"llo"
select s, b
```

通过转义符对双引号进行输出

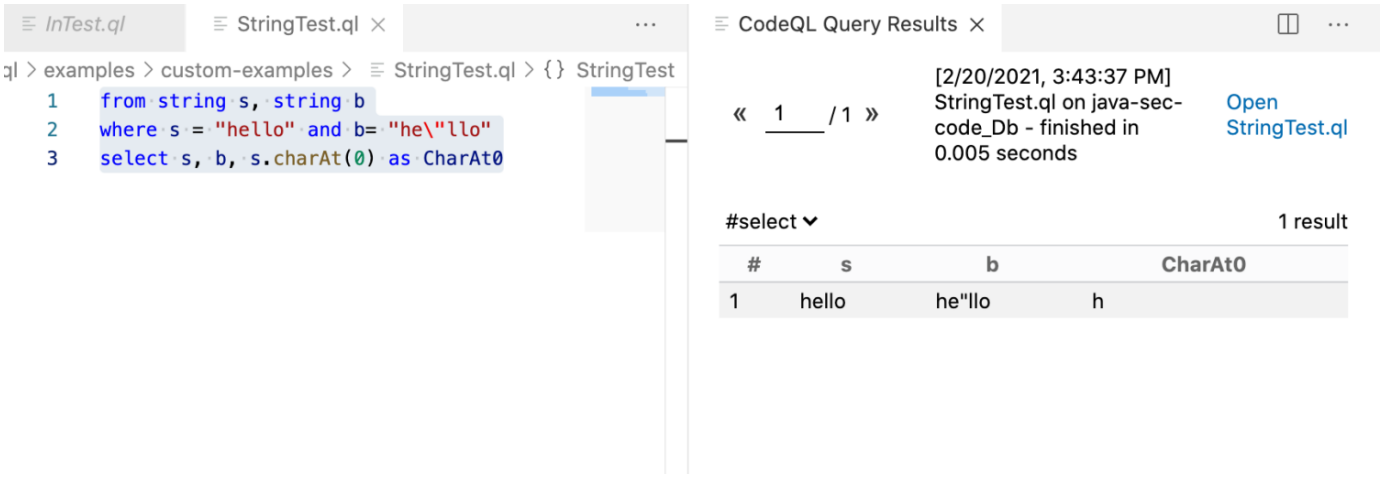
以下为常见的转义符

```
n  \" 表示字符"
n  \\ 表示字符\
n  \n 表示换行符
n  \r 表示回车符
n  \t 表示制表符
```

内建函数

QL语言还为字符串类型提供了许多内置的函数，按照官方的说法，就是内置谓词，例如charAt()函数，该函数可以接收一个表示字符串下标的整型参数，并返回指定下标处的字符。准确来说，该函数的返回值的类型仍然是字符串类型，只不过只包含单个字符而已。请看下面的示例代码

```
from string s, string b
where s = "hello" and b= "he\"llo"
select s, b, s.charAt(0) as CharAt0
```



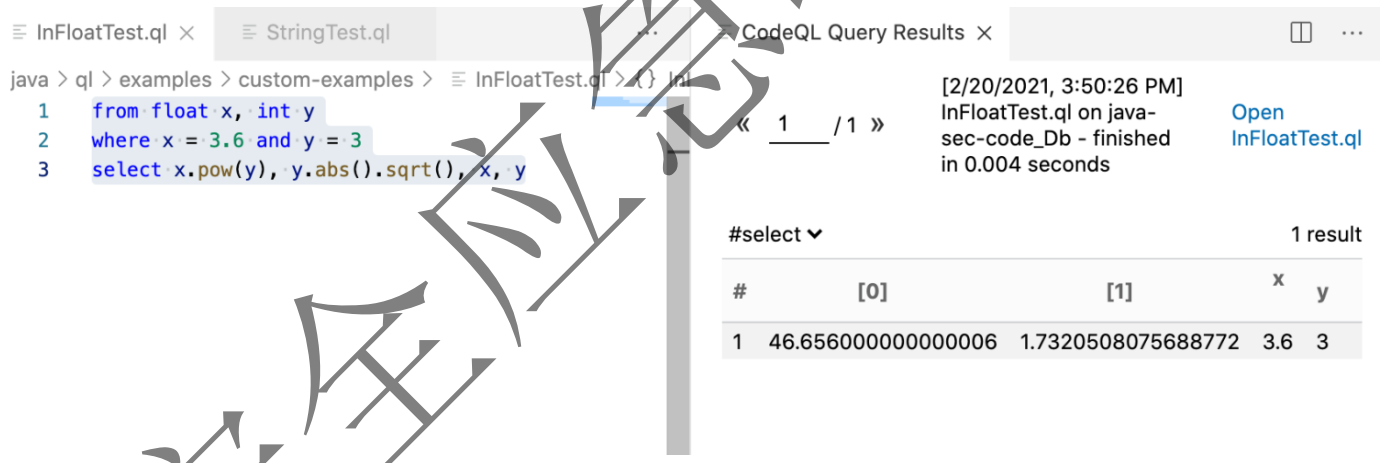
其他内建函数请参考QL的语言规范

整数以及浮点数

简单来说，整型变量用于保存整数，如306；而浮点型变量则用于保存浮点数，也就是带小数位的数，如3.14。例如：

```
from float x, int y
where x = 3.6 and y = 3
select x.pow(y), y.abs().sqrt(), x, y
```

就本例来说，上述代码实际上就是计算3.6的3次方，运行结果为：

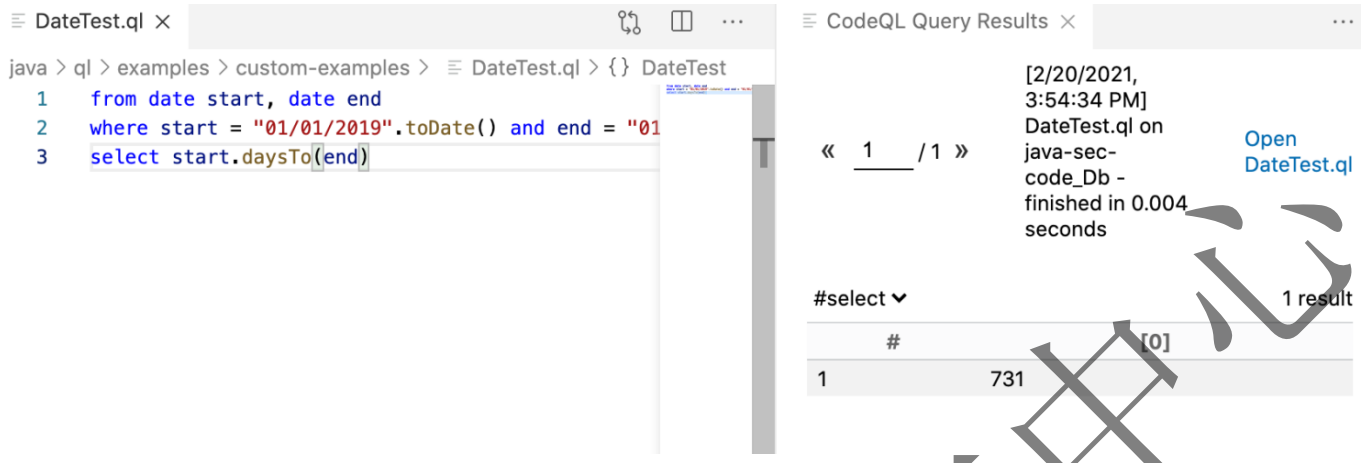


同样的，整型和浮点型也内建了许多函数，例如，abs()函数等，并且，它们的大部分函数的名称和作用都是相同的，感兴趣的读者可以参阅QL的语言规范

日期型

日期型变量用于保存公历表示的时间值和日期值，如年、月、日、时、分、秒以及毫秒等，注意，它们的取值都是整数。其中，表示年的整数的取值范围是从-16777216到16777215，表示月的整数的取值范围为从0到11，表示日的整数的取值范围是从1到31，表示时的整数的取值范围是从0到23，表示分的整数的取值范围是从0到59，表示秒的整数的取值范围是从0到59，表示毫秒的整数的取值范围是从0到999。

下面，我们编写一个查询，来计算从2021年1月1月到2019年1月1日为止已经过去了多少天了，具体如下所示：



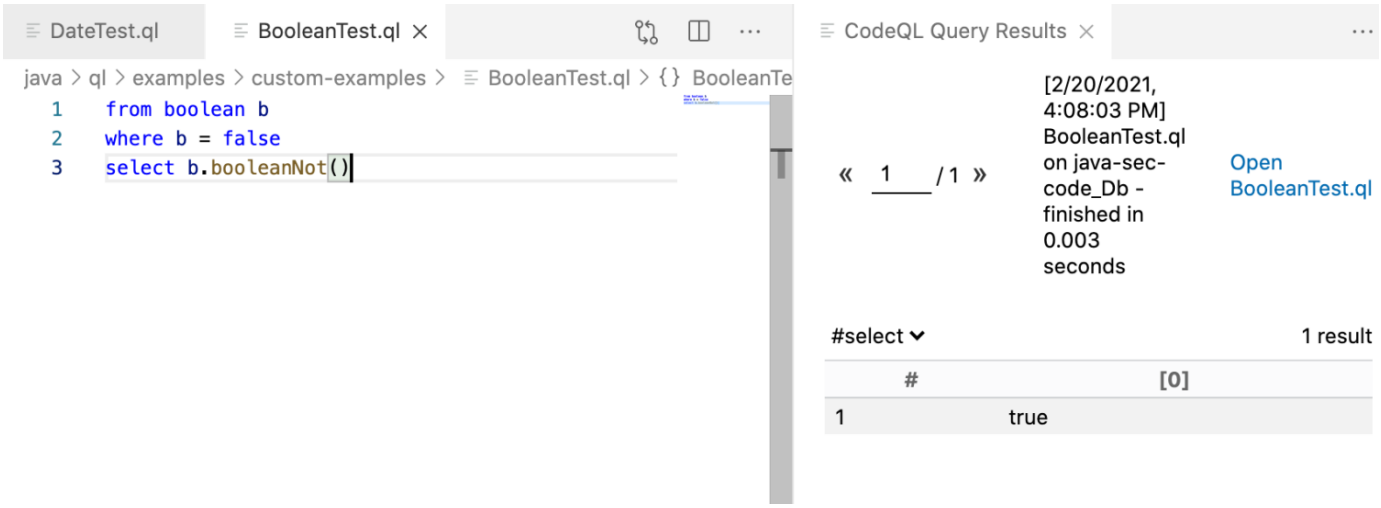
日期型的内建函数可以参考：QL的语言规范

布尔型

布尔型变量用来存放布尔值，即false（假）或者 true（真）。为了便于读者理解，这里举例说明：

```
from boolean b
where b = false
select b.booleanNot()
```

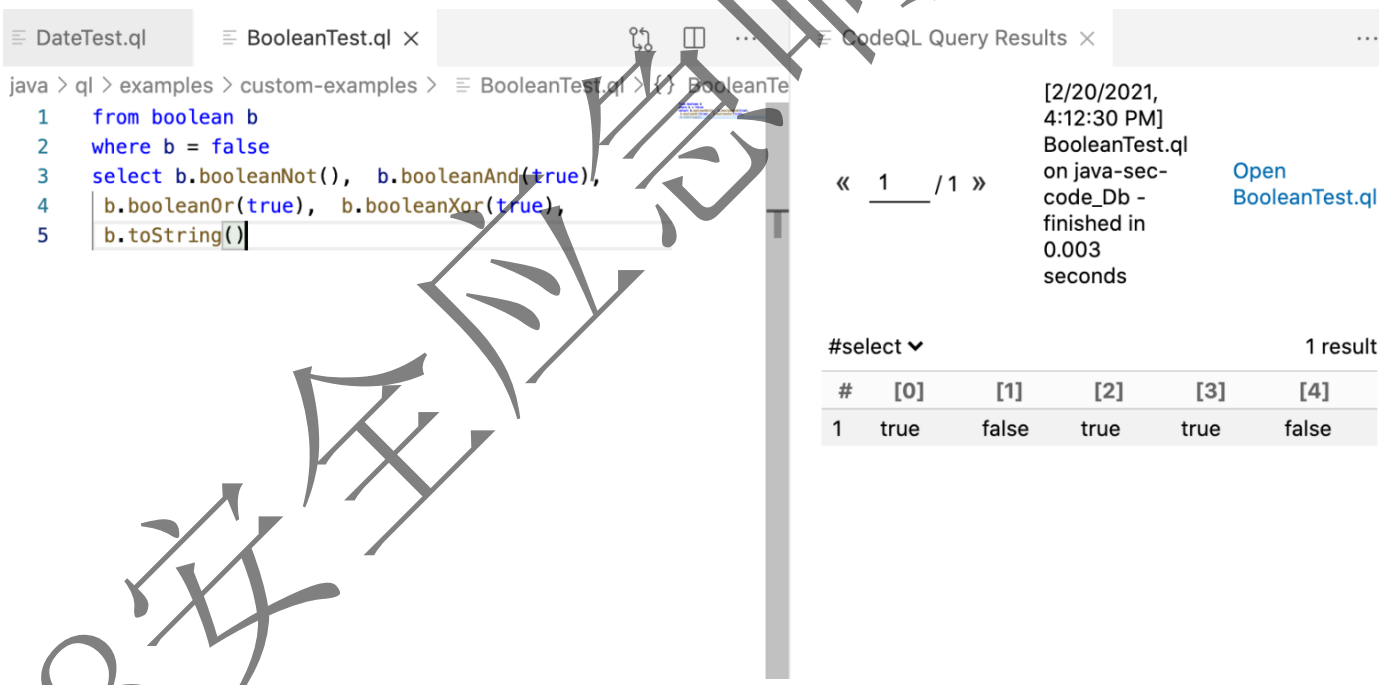
在上面的代码中，我们定义了一个布尔型变量b，并将其赋值为false，最后返回对变量b进行逻辑非操作后的值。上述代码的运行结果为：



在上面的代码中，我们定义了一个布尔型变量b，并将其赋值为false，最后返回对变量b进行逻辑非操作后的值。上述代码的运行结果为： 我们可以对布尔型变量b进行取反、以及and、or、Xor的逻辑运算，也可以将布尔型变量通过toString转换成String类型变量

```
from boolean b
where b = false
select b.booleanNot(),  b.booleanAnd(true),  b.booleanOr(true),  b
.booleanXor(true), b.toString()
```

运行结果：



小结

本文中我们介绍了QL语言的最基本的数据类型，并演示了如何编写和运行简单的QL程序