QByteArray在串口通讯中经常被使用

QByteArray类提供了很方便的对字节流操作的接口。

可以存储**raw bytes**和传统的**8-bits**的字符串，都是以**'\0'**结尾的，使用比**char***更方便，从串口读取到的**QByteArray**数据，一般需要进行提取和解析，此时就需要**QByteArray**转换为各类型数据。常用转换包括：

- 转为**HEX**，用于显示十六进制，这点在调试时特别有用，因为大多**HEX**码是没有字符显示的，如**0x00**、**0x20**等等；
- 转为不同进制数值并显示，如二进制、八进制、十进制和十六进制等；
- 转为整型、浮点型等数值类型；
- 大小写转换；
- 转为字符串类型；

QByteArray与QString区别

- **QString**是专门用来处理字符串的，除了能处理**ASCII**编码字符，还包括各国语言的编码，默认情况下**QString**会把所有数据当做**utf-8**编码来处理。

  QByteArray只是单纯用来处理数据的，除了能处理**ASCII**编码字符，其它复杂的编码不能处理（所以汉字不能处理），直接以字节流的方式来对待

```
QString str("小马哥");
QByteArray byte("小马哥");
qDebug() << "str:" << str << "byte:" << byte << endl;
//str: "小马哥" byte: "\xE5\xB0\x8F\xE9\xA9\xAC\xE5\x93\xA5"
```

- **str**保留编码格式，能输出中文，但是**QByteArray**只把**"**小马哥**"**当做普通的字节数据来处理，**utf-8**编码下，一个汉字占三个字节。

与字符串互转

- **QByteArray**与**QString**互转极为简单，二者从本质上类似，都是连续存储，区别是**QByteArray可以存无法显示的字符**，**QString只存可显示的字符**。如**QByteArray**可以存**0x00-0x19**，而**QString**则存储如**0x30**等可显示字符（**0x20-0x7E**）。可显示字符可参见**ASCII**表。
- **QString**也可以存储字符串信息，但通常以**16-bits**形式**Unicode**方式存储，这有利于非**ASCII**和非**Latin-1**格式的数据的存储

QByteArray转为QString：

```
QByteArray ba("123abc小马哥");
QString str = ba;
qDebug()<<"str:"<<str<<"ba:"<<ba ;
//输出: str:"123abc小马哥" ba:"123abc\xE5\xB0\x8F\xE9\xA9\xAC\xE5\x93\xA5"
```

QString转为QByteArray：

```
QString str("123abc小马哥");
QByteArray byte1 = str.toLatin1(); //按照ASCII编码转换，无法转换中文
QByteArray byte2 = str.toUtf8(); //按照Utf-8编码转换，可以转换中文
qDebug() <<  "byte1:" << byte1 << "byte2:" << byte2;
//byte1: "123abc???" byte2: "123abc\xE5\xB0\x8F\xE9\xA9\xAC\xE5\x93\xA5"
```

字符串数值转为各类数值

- **QByteArray**若为字符串数值，可通过**to**的方法转为各种类型数据

```
QByteArray string("1234.56");
qDebug() << string.toInt();
Debug() << string.toFloat();
qDebug() << string.toDouble();
```

数值转换与输出

- 尽管**QByteArray**是一个集合，但也可以作为一个特殊形式的数值用，其灵活的转换格式，可大大方便各种格式数据转换与显示的需求。如显示二进制和十六进制、显示科学计数和指定小数位四舍五入的数值

```
int n = 63;
qDebug()<<QByteArray::number(n); // returns "63"
qDebug()<<QByteArray::number(n, 16); // returns "3f"
qDebug()<<QByteArray::number(n, 16).toUpper(); // returns "3F"
qDebug()<<QByteArray::number(n, 2); // returns "111111"
qDebug()<<QByteArray::number(n, 8); // returns "77"
QByteArray::number(12.3456, 'f', 3); // returns "12.346"
```

Hex转换

- 把**Hex**编码转换为**char**存储到**QByteArray**，因为大多**HEX**码是没有字符显示的，如**0x00**、**0x20**等等；

```
QByteArray text = QByteArray::fromHex("517420697320677265617421");
text.data(); // returns "Qt is great!"
```

大小写转换

- **QByteArray**若为带大小写的字符串，可通过**toUpper()**和**toLower()**方法实现大小写转换