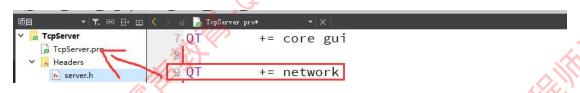
2033-Qt 网络编程体系--零声教育

一、Qt 网络编程基础

- 1、Qt 直接提供网络编程模块,基于 TCP/IP 客户端和服务器相关各种 类。 TCP 通信(QTcpSocket/QTcpServer)。 UDP 通信(QUdpSocket)。还有部分实现 HTTP、FTP 等网络协议的高级类。如 QNetworkRequest/QNetworkAccessManager等。
- 2、Qt 网络编程模块提供网络承载管理类,提供基于安全套接字层协议(SSL)的安全网络通信类。
- 3、我们开发过程中, UDP、TCP、HTTP 通信类等, 必须在程序中引用 Qt 网络模块, 项目配置文件的一条配置语句: QT +=network。



4、QHostInfo 类: QHostInfo 类为主机名查找提供静态函数,主要用来查询主机信息、包括主机名、ip 地址、DNS 域名等信息。

QHostInfo Class The QHostInfo class provides static functions for host name lookups. More... Header: #include <QHostInfo> qmake: QT += network • List of all members, including inherited members Note: All functions in this class are reentrant.

Public Types

enum HostInfoError { NoError, HostNotFound, UnknownError }

Public Functions

QHostInfo(int id = -1)

QHostInfo(const QHostInfo &other)

~QHostInfo()

QList<QHostAddress> addresses() const

QHostInfo::HostInfoError error() const

QString errorString() const

QString hostName() const

int lookupId() const

void setAddresses(const QList<QHostAddress> &addresses)

void setError(QHostInfo::HostInfoError error)

Static Public Members

void abortHostLookup(int id)

QHostInfo fromName(const QString &name)

QString localDomainName()

QString localHostName()

int lookupHost(const QString &name, QObject *receiver, const char *member)

int lookupHost(const QString &name, const QObject *receiver, PointerToMemberFunction function)

int lookupHost(const QString &name, Functor functor)

int lookupHost(const QString &name, const QObject *context, Functor functor)

5、QNetworkInterface 类,主要用于获取主机所有 IP 地址和网络

接口列表信息。具体常用函数如下:

QNetworkInterface Class

The QNetworkInterface class provides a listing of the host's IP addresses and network interfaces. More...

Header: #include <QNetworkInterface>

qmake: QT += network

Since: Qt 4.2

List of all members, including inherited members

Note: All functions in this class are reentrant.

Public Functions

QNetworkInterface()

QNetworkInterface(const QNetworkInterface &other)

~QNetworkInterface()

QList<QNetworkAddressEntry> addressEntries() const

QNetworkInterface::InterfaceFlags flags() const

QString hardwareAddress() const QString humanReadableName() const

int index() const

bool isValid() const

int maximumTransmissionUnit() const

QString name() const

roid swap(QNetworkInterface & other)

QNetworkInterface::InterfaceType type() const

QNetworkInterface & operator=(QNetworkInterface &&other)

QNetworkInterface & operator=(const QNetworkInterface & other)

Static Public Members

QList<QHostAddress> allAddresses()
QList<QNetworkInterface> allInterfaces()

QNetworkInterface interfaceFromIndex(int index)

QNetworkInterface interfaceFromName(const QString &name)

nt interfaceIndexFromName(const QString &name)

QString interfaceNameFromIndex(int index)

const QMetaObject staticMetaObject

二、【程序运行结果】

1、获取本地主机名称和 IP 地址 与 本地机器详细信息。

		1/m
主机名称:	Microsoft-Vico	获取主机名称和IP地址
本机IP地址:	192, 168, 1, 31	获取本地 <mark>机器详细</mark> 信息

■ 获取主机名称和IP--与--本地机器详细信息

主机名称:

Microsoft-Vico

获取主机名称和IP地址

本机IP地址:

192. 168. 1. 31

获取本地机器详细信息

■ 数据

0

设备名称: ethernet_32773 硬件地址: 90:2E:16:C9:09:D5

IP 地址: fe80::6082:b3ba:9409:2878%ethernet_32773

子网掩码: ffff:ffff:ffff:

广播地址:

IP 地址: 192.168.1.31 子网掩码: 255.255.255.0 广播地址: 192.168.1.255

设备名称: wireless_32768 硬件地址: 20:1E:88:CF:EB:94

IP 地址: fe80::956d:3701:98f0;2c26%wireless_32768

子网掩码: ffff:ffff:ffff:;;;

广播地址:

IP 地址: 169.254.44.38 子网掩码: 255.255.0.0 广播地址: 169.254.255.255

设备名称: wireless_32769 硬件地址: 20:1E:88:CF:EB:95

IP 地址: fe80::29f1:ed9e:ae0a:6ffb%wireless_32769

子网掩码: ffff:ffff:ffff:ffff:

广播地址:

IP 地址: 169.254.111.251 子网掩码: 255.255.0.0 广播地址: 169.254.255.255

设备名称: wireless_32770 硬件地址: 22:1E:88:CF:EB:94

IP 地址: fe80::4d75:de06:159f:1d68%wireless 32770

子网掩码: ffff:ffff:ffff:

广播地址:

IP 地址: 169.254.29.104 子网掩码: 255.255.0.0 广播地址: 169.254.255.255 A THE STATE OF THE

是排門

XXXXX

大场人提供的

2、查询域名或 IP 地址。



