

# Working with QSystemInfo - System Information API - Part 1

<b>ID</b>		<b>Creation date</b>	8-March-2010
<b>Platform</b>	S60 3rd Edition, FP1, FP2 S60 5th Edition	<b>Tested on devices</b>	Nokia 5800 xPressMusic
<b>Category</b>	Qt	<b>Subcategory</b>	Qt Mobility API

**Keywords (APIs, classes, methods, functions):** QSystemInfo, QString, QStringList, QPushButton



Tip:

- Please read this article before moving forward: [Archived:Setting up environment for Qt Mobility API](#)
- Setup Carbide.c++ IDE as explained in this article: [Working with Carbide.c++ IDE for Qt Mobility APIs](#)

## Overview

This is a part 1 of the **QSystemInfo API** which is a part of the **System Information API** from **Qt Mobility project**.

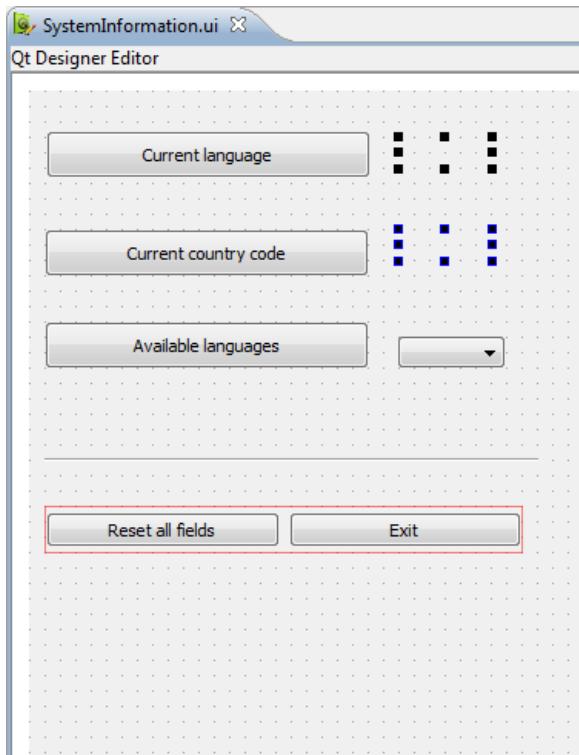
## What is covered in QSystemInfo API Part 1

- It is a part of the System Information API.
- This article covers information about
  - Available languages in the device
  - Current language of the device
  - Current country code of the device

## Capabilities required

- None

## UI design (.ui file)



## Project configuration file (.Pro file)

- Add the Qt Mobility project configuration option in the .Pro file as shown below

```
CONFIG += mobility
MOBILITY += systeminfo
```

- Add Qt-specific configuration options in .pro file as shown below

```
QT += network
```

## Implementing QSystemInfo API

- QSystemInfo API is a part of the QtMobility namespace. So declare the QtMobility namespace as shown below:

```
using namespace QtMobility;
```

## Header file

```
#ifndef SYSTEMINFORMATION_H
#define SYSTEMINFORMATION_H

#include <QtGui/QWidget>
#include "ui_SystemInformation.h"
#include <qsysteminfo.h>

using namespace QtMobility;
```

```
class SystemInformation : public QWidget
{
    Q_OBJECT

public:
    SystemInformation(QWidget *parent = 0);
    ~SystemInformation();

public slots:
    void displayCurrentCountryCode();
    void displayCurrentLanguage();
    void displayAvailableCountryCodes();
    void resetAllFields();

private:
    Ui::SystemInformation ui;
    QSystemInfo* sysInfo;
};

#endif // SYSTEMINFORMATION_H
```

## Source file

```
#include "SystemInformation.h"

SystemInformation::SystemInformation(QWidget *parent)
    : QWidget(parent), sysInfo(NULL)
{
    ui.setupUi(this);
    sysInfo = new QSystemInfo();

    QObject::connect(ui.curLanguageButton,
                     SIGNAL(clicked()),
                     this,
                     SLOT(displayCurrentLanguage()));
    QObject::connect(ui.countryCodeButton,
                     SIGNAL(clicked()),
                     this,
                     SLOT(displayCurrentCountryCode()));
    QObject::connect(ui.availableLanguagesButton,
                     SIGNAL(clicked()),
                     this,
                     SLOT(displayAvailableCountryCodes()));
    QObject::connect(ui.resetButton,
                     SIGNAL(clicked()),
                     this,
                     SLOT(resetAllFields()));
}

SystemInformation::~SystemInformation()
{
    delete sysInfo;
    sysInfo = NULL;
}
```

```
void SystemInformation::displayCurrentCountryCode()
{
    if(sysInfo)
    {
        QString str = sysInfo->currentCountryCode();
        ui.countryCodeLabel->setText(str);
    }
}

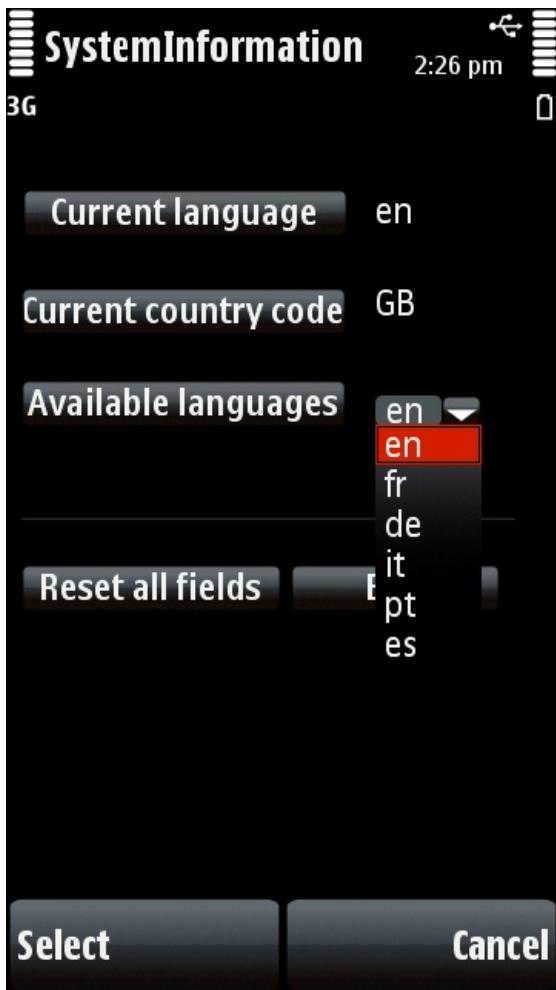
void SystemInformation::displayCurrentLanguage()
{
    if(sysInfo)
    {
        QString str = sysInfo->currentLanguage();
        ui.languageLabel->setText(str);
    }
}

void SystemInformation::displayAvailableCountryCodes()
{
    if(sysInfo)
    {
        QStringList list = sysInfo->availableLanguages();
        ui.languageComboBox->addItems(list);
    }
}

void SystemInformation::resetAllFields()
{
    if(sysInfo)
    {
        ui.languageLabel->clear();
        ui.countryCodeLabel->clear();
        ui.languageComboBox->clear();
    }
}
```

## Output

---



## Useful functions

---

- `QStringList availableLanguages();`
- `QString currentCountryCode();`
- `QString currentLanguage();`

## Keywords

---

### Headers

---

- `#include <qsysteminfo.h>`

### Classes

---

- `QSystemInfo`
- `QString`
- `QStringList`
- `QPushButton`

## Example Application

---

- A working example application is available to download from here: [QSystemInfo part1.zip](#)

## Related articles

---

- [Getting started with Qt Mobility APIs](#)
- [Archived:Setting up environment for Qt Mobility API](#)

- [Working with Carbide.c++ IDE for Qt Mobility APIs](#)
- [Working with QSystemInfo - System Information API - Part 2](#)
- [Working with QSystemDeviceInfo - System Information API - Part 1](#)
- [Working with QSystemStorageInfo - System Information API](#)
- [Working with QSystemDisplayInfo - System Information API](#)
- [Working with QSystemNetworkInfo - System Information API - Part 1](#)
- [Working with QSystemNetworkInfo - System Information API - Part 2](#)

## Reference links

---

- [Qt - cross-platform application and UI framework](#)
- [Qt Mobility API](#)
- [New Qt APIs Beta - Mobility Project](#)
- [SDK help](#)