# qml Settings用于保存用设置户数据，当程序重新启动还能使用设置数据。要使用Settings必须做以下准备

## 1.需要导入：import Qt.labs.settings 1.0

## 2.在main.cpp文件在设置组织名称，组织位置，应用程序名称

#### 简单使用：

|  |  |
| --- | --- |
| //main.cpp  #include <QGuiApplication>  #include <QQmlApplicationEngine>  int **main**(int **argc**, char \***argv**[])  {  #if QT\_VERSION < QT\_VERSION\_CHECK(6, 0, 0)  QCoreApplication::setAttribute(Qt::AA\_EnableHighDpiScaling);  #endif  QGuiApplication **app**(*argc*, *argv*);  //设置组织名称，组织位置，应用程序名称  app.setOrganizationName("Rest El Rico");  app.setOrganizationDomain("http://www.google.com");  app.setApplicationName("Settingsdemo");  QQmlApplicationEngine **engine**;  const QUrl **url**(QStringLiteral("qrc:/main.qml"));  QObject::connect(&engine, &QQmlApplicationEngine::objectCreated,  &app, [url](QObject \***obj**, const QUrl &**objUrl**) {  if (!obj && url == objUrl)  QCoreApplication::exit(-1);  }, Qt::QueuedConnection);  engine.load(url);  return app.exec();  } | //main.qml  import QtQuick 2.15  import QtQuick.Window 2.15  import Qt.labs.settings 1.0  import QtQuick.Controls 2.15  Window {  width: 640  height: 480  visible: true  title: *qsTr*("Settings demo")  Settings{  id:*settings*  property int speed: 0  property string calledname: "fast speed"  }  Text {  x:200  y:80  id: *txt*  text: *settings*.speed  font.pixelSize: 40  }  Button{  text: "set"  onClicked: {  *settings*.speed +=100  }  anchors.bottom : *parent*.bottom  x:100  anchors.bottomMargin: 30  }  } |

效果：

|  |  |  |
| --- | --- | --- |
| //初始状态 | //点击set修改值，关闭窗口 | //重新启动重新 |

#### 以上的功能是qml自动完成的。我们也可以手动将设置保存到一个我们指定的文件中，使用的是Settings的fileName属性

|  |  |
| --- | --- |
| import QtQuick 2.15  import QtQuick.Window 2.15  import Qt.labs.settings 1.0  import QtQuick.Controls 2.15  Window {  width: 640  height: 480  visible: true  title: *qsTr*("Settings demo")  Settings{  id:*settings*  property int speed: 0  property string calledname: "fast speed"  fileName: "setting.ini"  }  Text {  x:200  y:80  id: *txt*  text: *settings*.calledname + ": "+*settings*.speed  font.pixelSize: 40  }  Button{  text: "set"  onClicked: {  *settings*.speed +=100  }  anchors.bottom : *parent*.bottom  x:100  anchors.bottomMargin: 30  }  } | 左边的代码一运行，就会在指定的路径下面生成一个settings.ini文件  内容大概是这样子的  [General]  speed=2300  calledname=fast speed  当我们每一次动态修改speed的值，都会将最新的值写到这个文件。  当应用程序重新启动就会到这个文件中读取数据并且显示到窗体 |

#### Settings的默认分组是General，我们可以设置它为别的分组，两个以上的Settings可以共享一个settings文件

|  |  |  |  |
| --- | --- | --- | --- |
| import QtQuick 2.15  import QtQuick.Window 2.15  import Qt.labs.settings 1.0  import QtQuick.Controls 2.15  Window {  width: 640  height: 480  visible: true  title: *qsTr*("Settings demo")  Settings{  id:*settings*  property int speed: 0  property string calledname: "fast speed"  fileName: "setting.ini"  }  Settings{  id:*settings2*  property int speed: 1200  property string calledname: "super speed"  fileName: "setting.ini"  category: "special"  }  Text {  x:200  y:80  id: *txt*  text: *settings*.calledname + ": "+*settings*.speed  font.pixelSize: 40  }  Text {  x:200  y:180  id: *txt2*  text: *settings2*.calledname + ": "+*settings2*.speed  font.pixelSize: 40  }  Button{  text: "set"  onClicked: {  *settings*.speed +=100  *settings2*.speed +=500  }  anchors.bottom : *parent*.bottom  x:100  anchors.bottomMargin: 30  }  } | |  | | --- | | //运行程序一轮狂点之后 | | 此时settings.ini的数据为：  [General]  speed=3000  calledname=fast speed  [special]  speed=4700  calledname=super speed | |

Settings有一个记住value()的方法，可以获取我们自己希尔settings.ini的属性值，比如我们在[General]节点（对应的是id为settings的Settings）下面新添加一个title属性值是：super speed pk

我们开始在Window的Component.onCompleted事件中读取这个title并且将Window的标题设置为这个值

|  |  |
| --- | --- |
| //settings.int的新内容  [General]  speed=3000  calledname=fast speed  title=super speed pk  [special]  speed=4700  calledname=super speed | //main.qml  import QtQuick 2.15  import QtQuick.Window 2.15  import Qt.labs.settings 1.0  import QtQuick.Controls 2.15  Window {  width: 640  height: 480  visible: true  title: *qsTr*("Settings demo")  Settings{  id:*settings*  property int speed: 0  property string calledname: "fast speed"  fileName: "setting.ini"  }  Settings{  id:*settings2*  property int speed: 1200  property string calledname: "super speed"  fileName: "setting.ini"  category: "special"  }  Text {  x:200  y:80  id: *txt*  text: *settings*.calledname + ": "+*settings*.speed  font.pixelSize: 40  }  Text {  x:200  y:180  id: *txt2*  text: *settings2*.calledname + ": "+*settings2*.speed  font.pixelSize: 40  }  Button{  text: "set"  onClicked: {  *settings*.speed +=100  *settings2*.speed +=500  }  anchors.bottom : *parent*.bottom  x:100  anchors.bottomMargin: 30  }  Component.onCompleted: {  *title* = *settings*.value("title","no data")//第二个参数是如果没有找到这个key就输出它的值  }  } |

效果：

|  |
| --- |
|  |

Settings有一个叫做setValue的方法可以添加属性设置