

▪ 第1209篇 • 图腾

▪ 评论 |

▪ 肖鹰：柴静的意义——她为何值得

▪ “深圳机场撞人事件”不是一个人

▪ 谁才是莫斯科刺杀事件的最大受益

▪ 第1207篇 • 日货

▪ 北美崔哥：美国正称霸世界，春晚

▪ 某些国家为何拿高铁来戏弄中国？

▪ 第1202篇 • 冤死

▪ 北美崔哥：中国式上访，已正式输

查看更多>>

谁看过这篇博文

swinna

3月2日

相逢一笑

2月28日

风xx摇叶

2月25日

右右youyou

2月24日

花丿开、…

2月7日

121984794

2月4日

odin

2月4日

重来

2月4日

积木38026…

1月30日

Blyne

1月30日

生活如进…

1月23日

常歌是我

1月22日

```
{
    return ;
}

//设置Home、End键直接将光标位置设到最前、最后，而不被选中
else if(event->key() == Qt::Key_Home)
{
    this->setCursorPosition(0);
}
else if(event->key() == Qt::Key_End)
{
    this->setCursorPosition(this->text().length());
}
else
{
    QLineEdit::keyPressEvent(event);
}
}

//鼠标移动时，什么都不干，因为一般情况下当鼠标移动文本都会被选中
void PwdLineEdit::mouseMoveEvent(QMouseEvent *event)
{
    return ;
}

//同上，鼠标双击什么也不干
void PwdLineEdit::mouseDoubleClickEvent(QMouseEvent *event)
{
    return ;
}

//这个事件比较重要，当按下Shift键时，点击文本是会被选中的，所以需要重新定义
void PwdLineEdit::mousePressEvent(QMouseEvent *event)
{
    //获得当前鼠标位置
    int cursor_pos = cursorPositionAt(event->pos());
    //获取光标位置
    int current_pos = this->cursorPosition();
    //鼠标点击位置在光标位置之前，则设置光标前移一格
    if(cursor_pos < current_pos)
    {
        this->setCursorPosition(current_pos - 1);
    }
    //鼠标点击位置在光标位置之后，则设置光标后移一格
    else if(cursor_pos > current_pos)
    {
        this->setCursorPosition(current_pos + 1);
    }
    else if(cursor_pos == current_pos)
    {
        this->setCursorPosition(current_pos);
    }
    else
    {
        QLineEdit::mousePressEvent(event);
    }
}
```

完工之后，全选、Home、End...都被屏蔽掉了，这样文本就在不被选中了！

但是如果存在其他组件的时候，上、下键还是会选中，所以再添加如下代码就行了！

```
bool LoginDialog::eventFilter(QObject *obj, QEvent *event)
{
    if(obj == domain_line_edit || obj == user_line_edit || obj == password_line_edit || obj ==
check_box ||
    obj == login_button || obj == forgot_pwd_button || obj == regist_button || obj == set_button ||
obj == min_button || obj == close_button)
    {
        if(event->type() == QEvent::KeyPress)
        {
            //将事件转化为键盘事件
            QKeyEvent *key_event = static_cast(event);

            bool domain_edit_focus = domain_line_edit->hasFocus();
            bool user_edit_focus = user_line_edit->hasFocus();
            bool password_edit_focus = password_line_edit->hasFocus();
            bool check_box_focus = check_box->hasFocus();
            bool set_btn_focus = set_button->hasFocus();
            bool login_btn_focus = login_button->hasFocus();
            bool get_pwd_btn_focus = forgot_pwd_button->hasFocus();
            bool register_btn_focus = regist_button->hasFocus();
            bool min_btn_focus = min_button->hasFocus();
            bool close_btn_focus = close_button->hasFocus();

            //按下Tab键执行焦点切换事件
            if(key_event->key() == Qt::Key_Tab)
            {
                if(domain_edit_focus)
                {
                    focusNextChild();
                    user_line_edit->setFocus();
                }
                else if(user_edit_focus)
                {
                    focusNextChild();

                    //获取密码框输入内容长度，并设置光标位置
                    int length = password_line_edit->text().length();
                    password_line_edit->setSelection(length, length);
                    password_line_edit->setFocus();
                }
                else if(password_edit_focus)
                {
                    focusNextChild();
                    check_box->setFocus();
                }
                else if(check_box_focus)
                {
                    focusNextChild();
                    login_button->setFocus();
                }
                else if(login_btn_focus)
                {
                    focusNextChild();
                    forgot_pwd_button->setFocus();
                }
                else if(get_pwd_btn_focus)
                {
                    focusNextChild();
                    regist_button->setFocus();
                }
                else if(register_btn_focus)
                {
                    focusNextChild();
                    set_button->setFocus();
                }
                else if(set_btn_focus)
                {
                    focusNextChild();
```

```
min_button->setFocus();
}
else if(min_btn_focus)
{
    focusNextChild();
    close_button->setFocus();
}
else if(close_btn_focus)
{
    focusNextChild();
    domain_line_edit->setFocus();
}
else
{
    focusNextChild();
}
return true;
}

//主要禁用上、下按键
else if(key_event->key() == Qt::Key_Up)
{
    return true;
}
else if(key_event->key() == Qt::Key_Down)
{
    return true;
}

//左右键功能，若为密码框则前进、后退一位，若为普通输入框，则设为默认操作，其他按钮，复选框则不起作用
else if(key_event->key() == Qt::Key_Left)
{
    if(domain_edit_focus || user_edit_focus)
    {
        return QDialog::eventFilter(obj, event);
    }
    else if(password_edit_focus)
    {
        if(password_line_edit->cursorPosition() > 0)
        {
            password_line_edit->setCursorPosition(password_line_edit->cursorPosition() - 1);
        }

        return true;
    }
    else
    {
        return true;
    }
}
else if(key_event->key() == Qt::Key_Right)
{
    if(domain_edit_focus || user_edit_focus)
    {
        return QDialog::eventFilter(obj, event);
    }
    else if(password_edit_focus)
    {
        if(password_line_edit->cursorPosition() < password_line_edit->text().length())
        {
            password_line_edit->setCursorPosition(password_line_edit->cursorPosition() + 1);
        }

        return true;
    }
    else
    {
        return true;
    }
}
```

```
    }  
    }  
    }  
  
    return QDialog::eventFilter(obj, event);  
}  
  
return QDialog::eventFilter(obj, event);  
}
```

这些就可以屏蔽掉文本内容被选中了，当然在多个控件切换的时候还有可能其他控件的切换会聚焦到密码框中，只需要像上面那样屏蔽掉就行了！

注：
技术在于交流、沟通，转载请注明出处并保持作品的完整性。
作者：`☆奋斗ing♥孩子` 原文：http://blog.sina.com.cn/s/blog_a6fb6cc90101b3b8.html。

3

👍喜欢

分享：       

阅读 (1500) | 评论 (0) | 收藏 (0) | 已有6人转载▼ | 喜欢▼ | 打印

已投稿到： 排行榜

前一篇：Qt之Esc键
后一篇：cin详解（cin.get()、cin.getline()、cin.clear()、cin.sync()）

评论









重要提示：警惕虚假中奖信息

[发评论]

做第一个评论者吧！ 抢沙发>>

发评论

一去、二三里：

☐  分享到微博 

☐ 匿名评论

验证码： 请点击后输入验证码 收听验证码

发评论

以上网友发言只代表其个人观点，不代表新浪网的观点或立场。

< 前一篇
Qt之Esc键

后一篇 >
cin详解（cin.get()、cin.getline()、cin.clear()、cin.sync()）

新浪BLOG意见反馈留言板 不良信息反馈 电话：4006900000 提示音后按1键（按当地市话标准计费） 欢迎批评指正

新浪简介 | About Sina | 广告服务 | 联系我们 | 招聘信息 | 网站律师 | SINA English | 会员注册 | 产品答疑

Copyright © 1996 - 2014 SINA Corporation, All Rights Reserved

新浪公司 版权所有