گزارش هفته اول آزمایشگاه آز سخت‌افزار گروه ۵(و۱)

# نصب Qt:

در ابتدا به نصب Qt می‌پردازیم مراحل آن برای لینوکس/مک و ویندوز مشابه می‌باشد و این توضیحات بر پایه‌ی نصب بر روی سیستم‌عامل لینوکس می‌باشد:

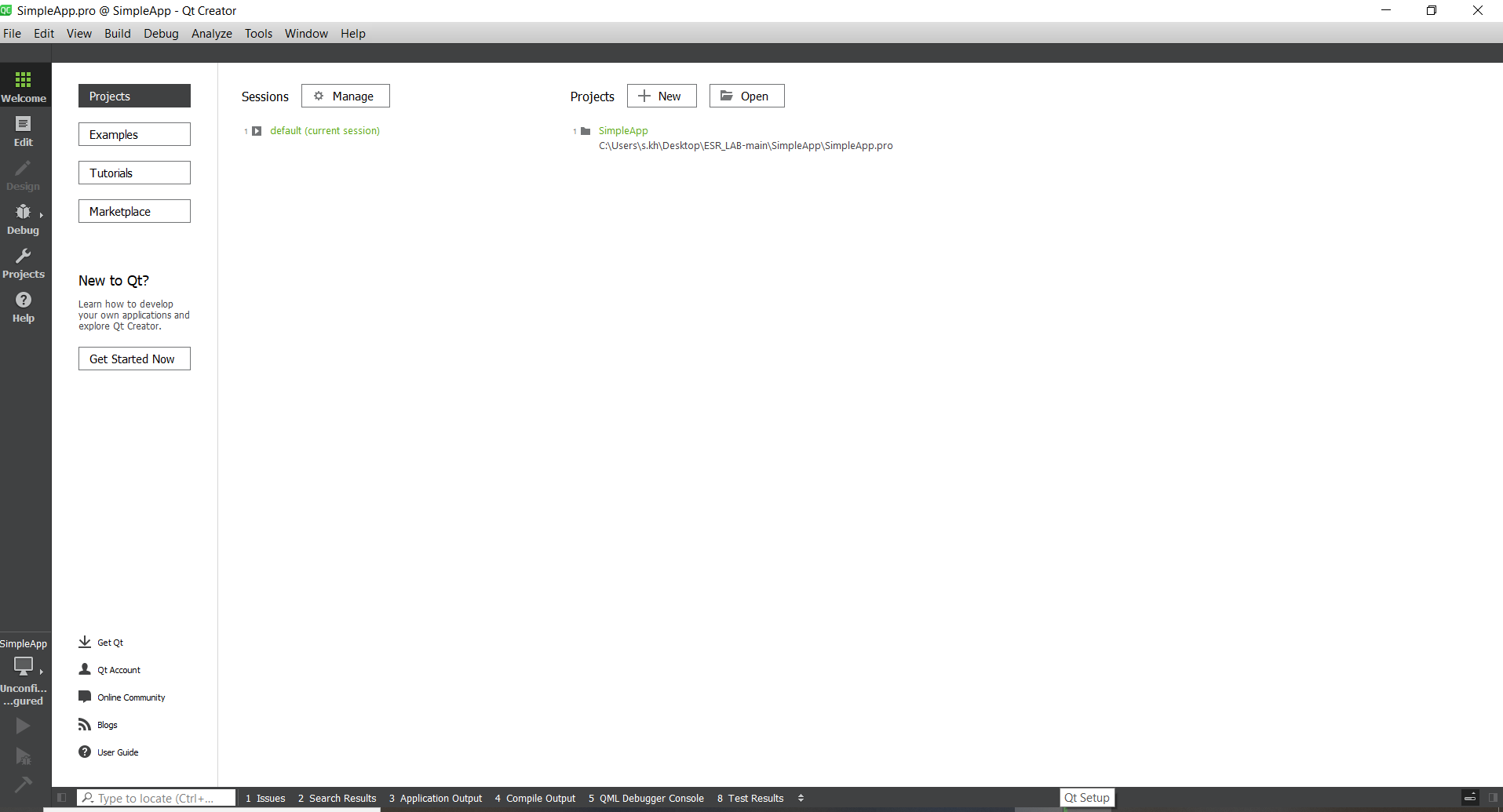
پس از ساختن اکانت در سایت qt.io از طریق [این لینک](https://www.qt.io/download-qt-installer?hsCtaTracking=99d9dd4f-5681-48d2-b096-470725510d34%7C074ddad0-fdef-4e53-8aa8-5e8a876d6ab4) فایل نصب آنلاین را دانلود کردیم. یک فایل .run با حجم 23 مگابایت دانلود شد که با دستور زیر آن‌ را قابل اجرا کردیم:

chmod +x <name>.run

سپس فایل را اجرا کرده و با انتخاب ورژن 6.2 و نیز پکیج های مناسب به جهت نصب (از میان پکیجهای زیر)، حدود 7 گیگابایت از 15 گیگابایت کتابخانه مورد نیاز را نصب کردیم.



محیط نرم افزار پس از نصب:



# صفحه کلید مجازی Qt

صفحه کلید مجازی Qt یک فریم ورک شناخته شده برای برنامه‌نویسی فرانت‌اند است که قابلیت اجرا روی سیستم‌عامل‌های مختلف از جمله سیستم‌عامل من (ویندوز) را دارد. Qt5 که یک فریمورک ورودی است، دارای تابع‌های آماده‌ی بسیاری است که با استفاده از آن‌ها می‌توان برنامه‌های مختلفی را مبتنی بر زبانی مثل C++ نوشت.

**ویژگی‌های اصلی**

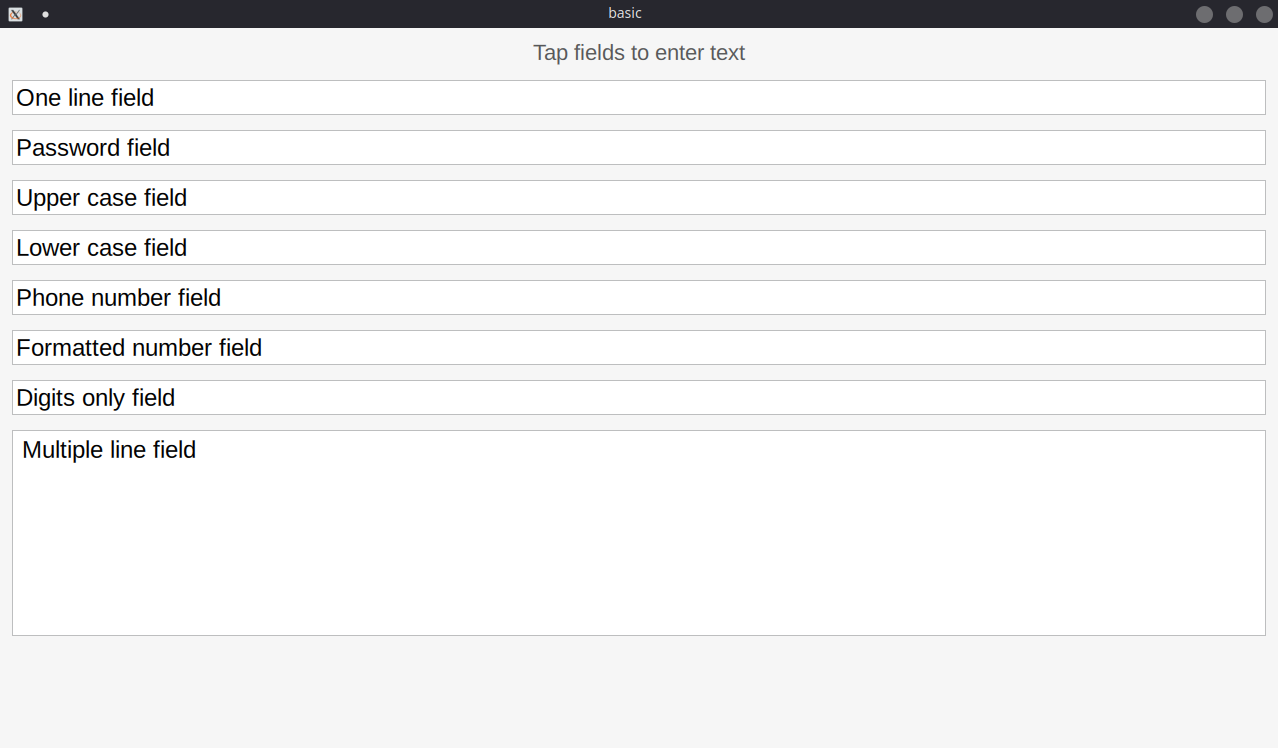
* پشتیبانی از سبک‌ها و استایل‌های مختلف صفحه‌کلید که به راحتی قابل تبدیل به یکدیگر هستند.
* پیش بینی خودکار متن
* تکمیل خودکار کلمه
* بزرگ سازی حرف اول در زبان انگلیسی به طور خودکار
* افزودن فاصله به صورت خودکار
* پشتیبانی از زبان‌های مختلف و کاراکترهای آن‌ها، برای مثال زبان انگلیسی، فارسی، عربی، چینی، ژاپنی، فرانسوی، ایتالیایی و ...
* پشتیبانی از هر دو استایل نوشتار (چپ به راست و راست به چپ)
* فهم نوشتار توسط صفحه کیبورد لمسی
* اجرا و توسعه بر روی پلتفرم‌های مختلف

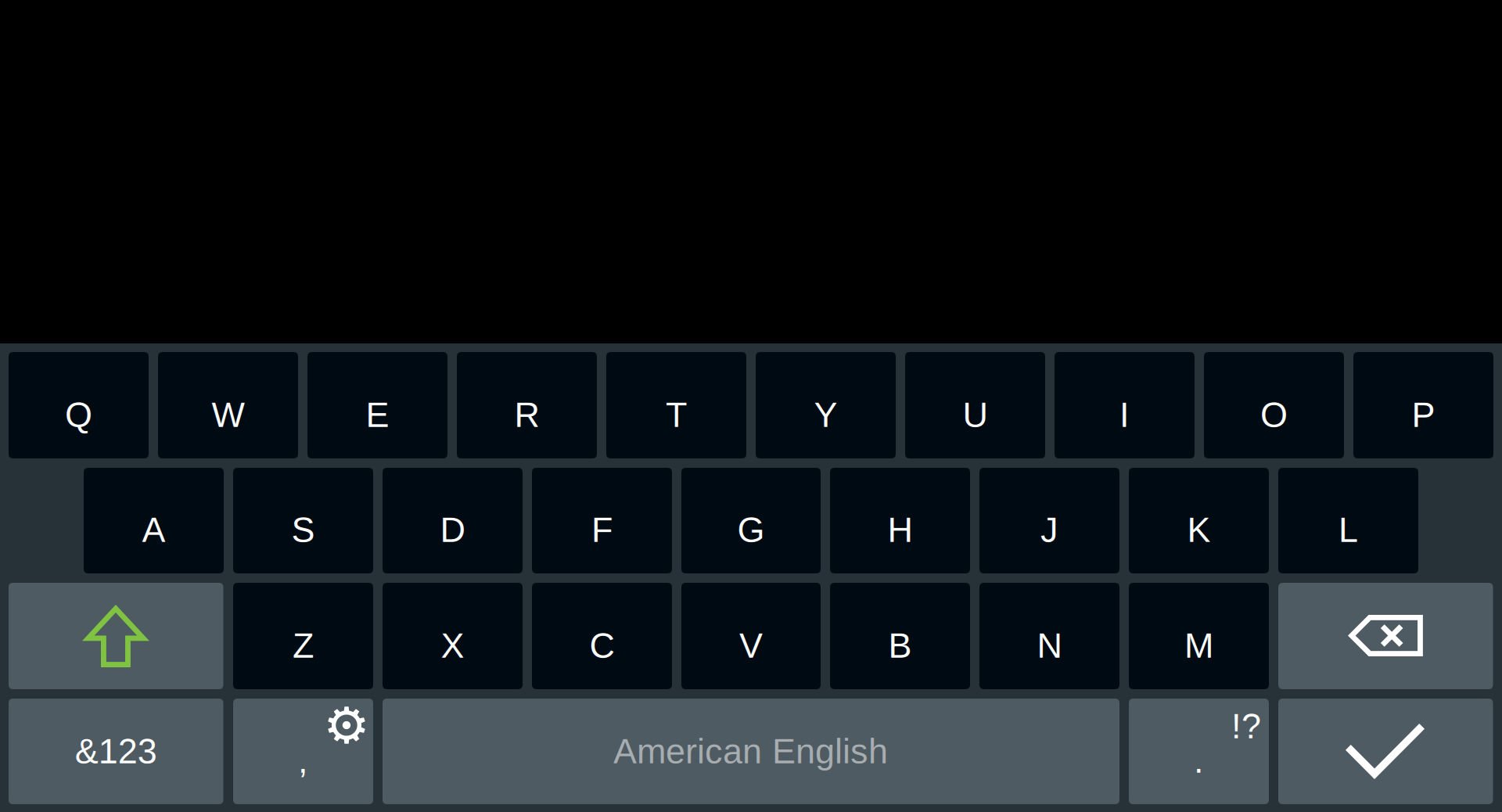
**پلاگین‌های Third-party**

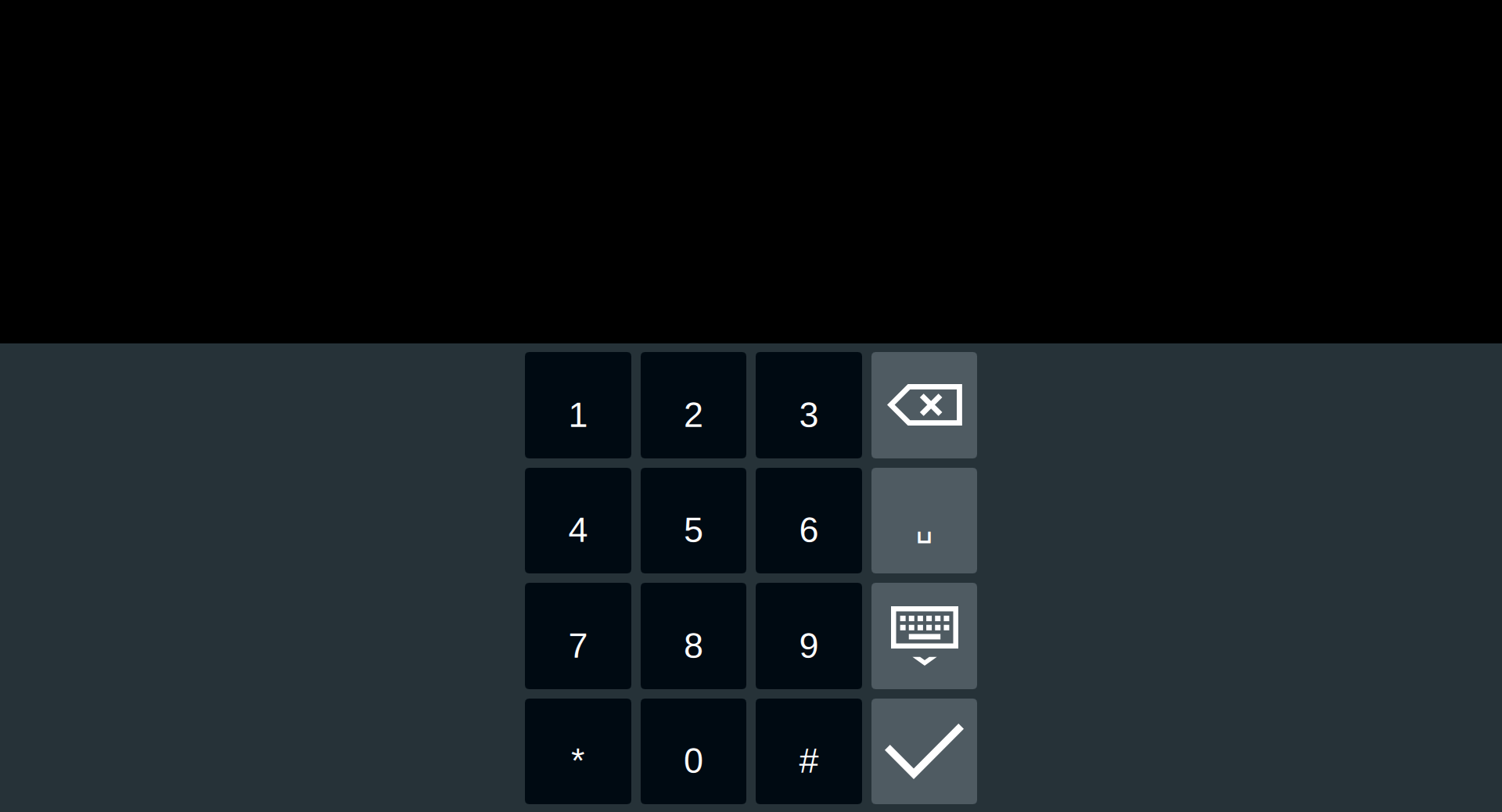
پلاگین‌هایی که قابلیت افزوده شدن به صفحه کلید مجازی Qt را دارند عبارتند از:

* T9 write: پشتیبانی از نوشتار دستنویس
* MyScript: تشخیص دستخط
* [Building Qt Virtual Keyboard](https://doc.qt.io/qt-5/qtvirtualkeyboard-build.html) دستورالعمل چگونگی استفاده از پلاگینهای مختلف در کنار یکدیگر

در نهایت کد سمپلی که در سایت Qt قرار داشت را اجرا کردیم که خروجی آن به صورت زیر می‌باشد:

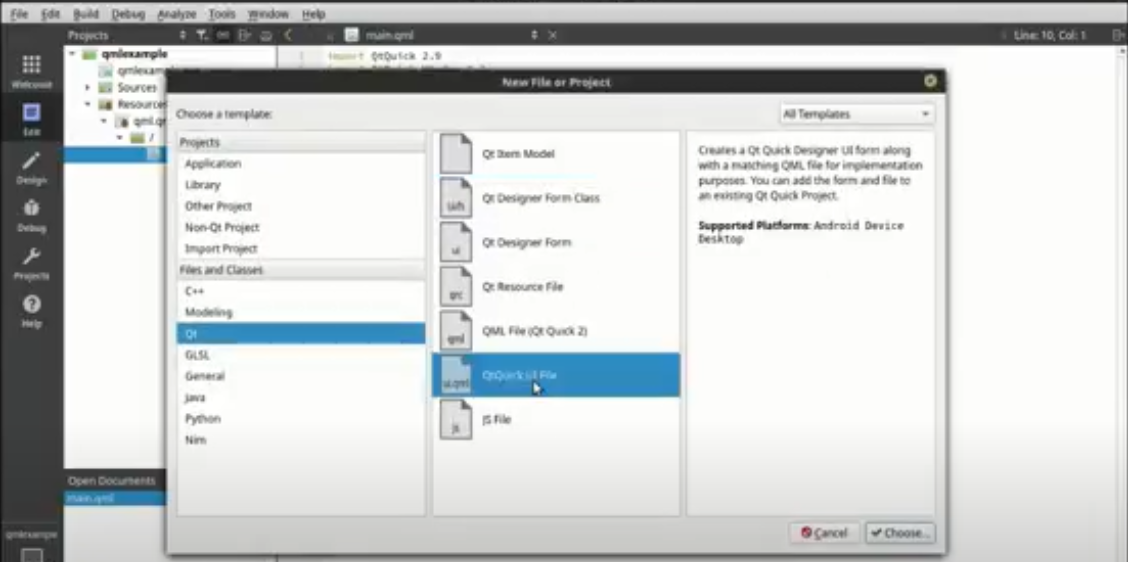
 و در صورت کلیک بر روی هر کدام از این فیلد‌ها با کیبورد‌های زیر رو‌به‌رو می‌شویم:



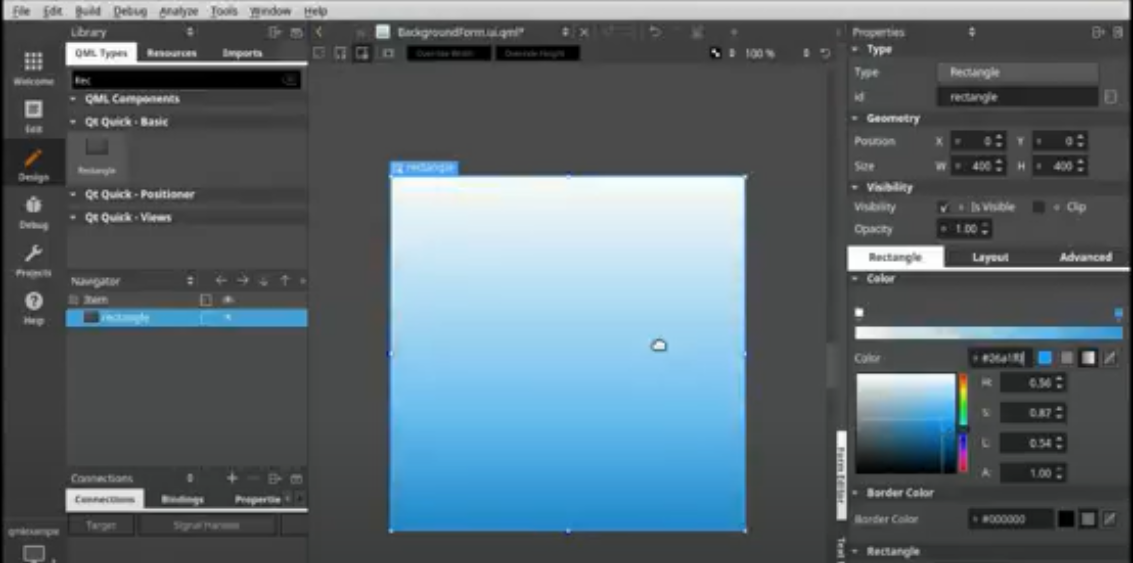


# ساختار QML:

برای توسعه ی یک اپلیکیشن در Qt، در دو بخش C++ و QML برنامه نوشته شده و سپس با اتصال این دو بخش به یکدیگر برنامه تکمیل میشود. بخش c++ هسته کد در QT را تشکیل میدهد و بخش QML حکم user interface یا همان UI را بازی میکند. در بخش QML میتوان اشیای مختلفی ساخت. برای مثال برای ایجاد یک مربع به عنوان بکگراند صفحه در این واسط گرافیکی، به صورت عمل میکنیم:

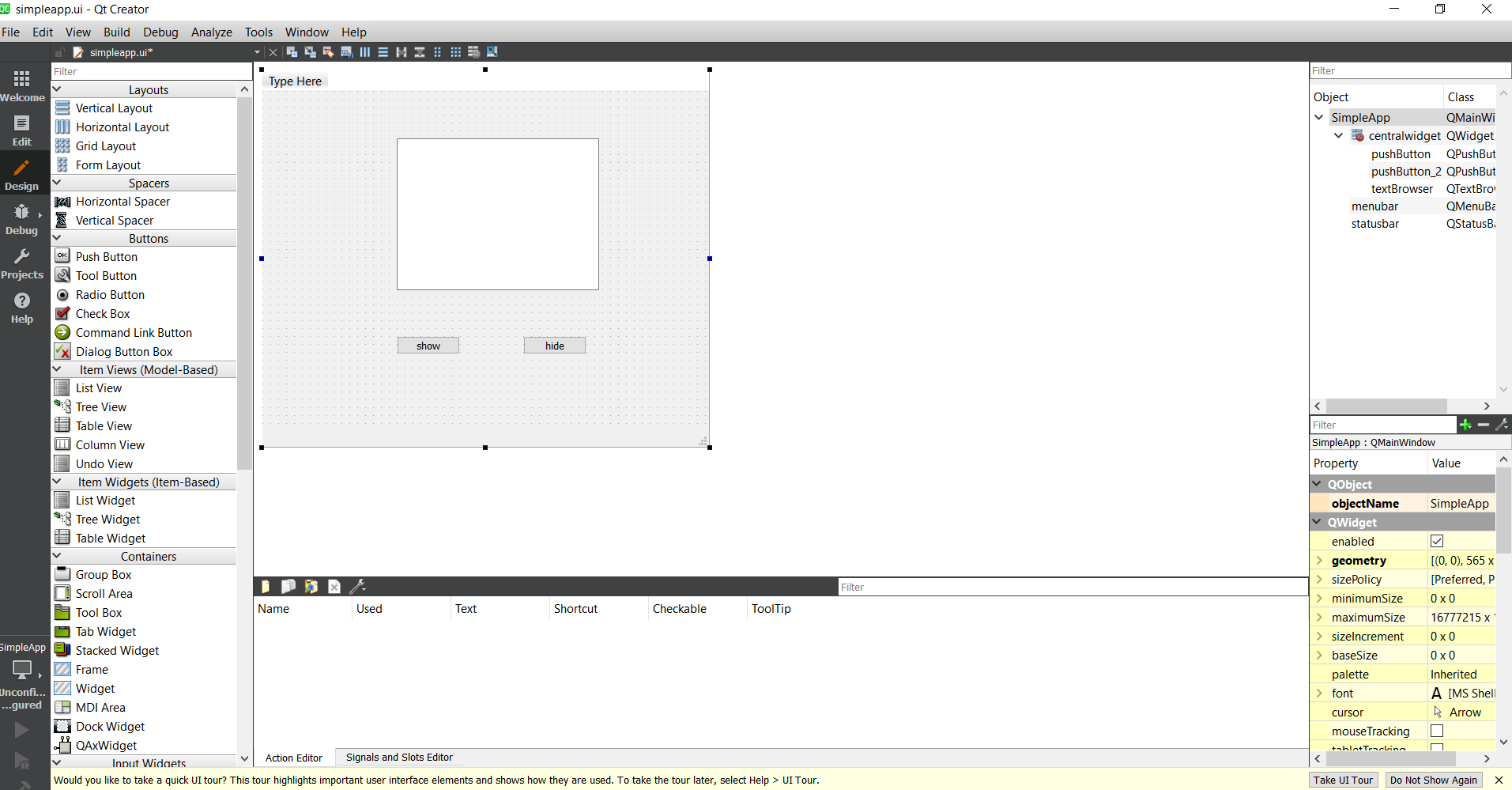


نام شی مورد نظرمان را انتخاب میکنیم (برای مثال اینجا نام آن را background در نظر میگیریم) و با رفتن به بخش تنظیمات، صفحه مدنظرمان را با سایز، رنگ و سایر مشخصات میسازیم:



به همین ترتیب سایر اشیا را نیز میتوانیم در این محیط ایجاد کنیم. به ازای هر شی ساخته شده در این محیط، یک سری کد نیز تولید میشود. زین پس ما با تغییر این کد یا مراجعه به صفحه پیشتر معرفی شده، میتوانیم به اشیای با تنظیمات دلخواه خود برسیم.

به طور کلی نیز در این صفحه می‌توانید سایر تنظیمات گرافیکی قابل دسترس در qt را مشاهده کنید:



حال برای استفاده از صفحه کلید مجازی در این واسط گرافیکی، ابتدا کتابخانه های زیر را import میکنیم:

QtQuick.VirtualKeyboard 2.2

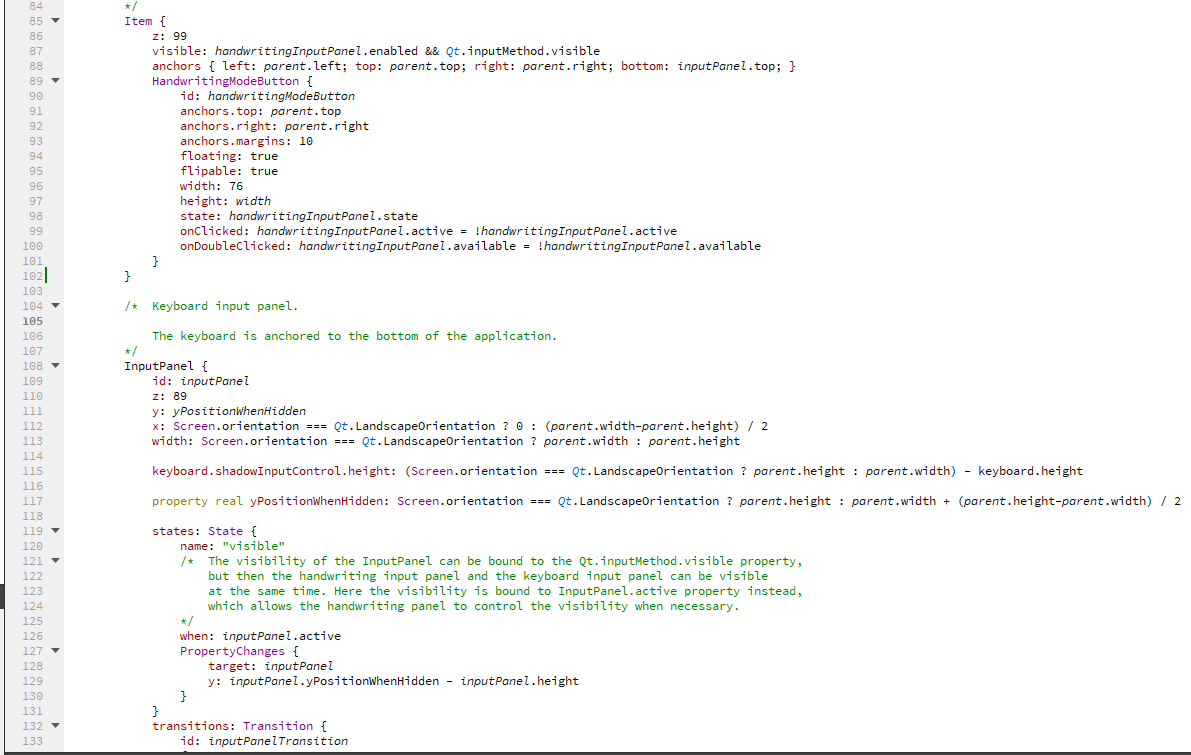
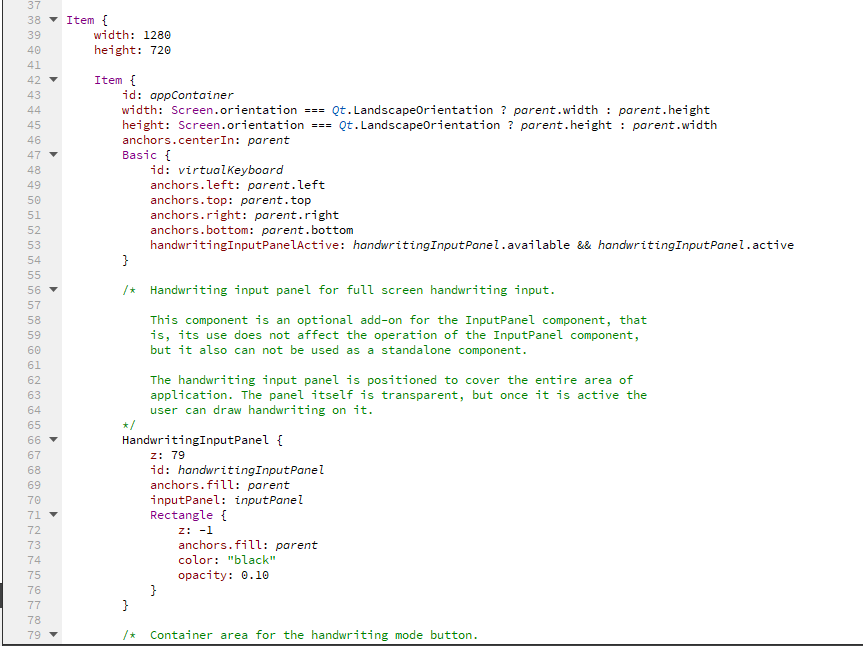
import QtQuick.VirtualKeyboard.Settings 2.2

QtQuick 2.0

QtQml 2.14

QtQuick.Window 2.2

میبینیم که در این بخش هر شی تحت عنوان یک Item قرار گرفته است. همچنین Handwriting Input Panel را مشاهده میکنیم که با آن کاربر میتواند با دستخط خود بنویسد و سامانه نهفته آن را دریافت نماید. این بخش نیز یک Item است. همچنین هر Item دارای مشخصاتی ظاهری از جمله id، رنگ، نودهای پدر و جد و ...، شفافیت، پنل ورودی، طول و عرض و در صورت لزوم ارتفاع، متن و سایر مشخصاتی که در صورت دریافت یک رخداد نیاز خواهند بود، است.



# Rectangle:

ویژگی‌های Item را به ارث می‌برد و با استفاده از آن می‌توان یک مستطیل را در صفحه قرار داد. ویژگی‌های مورد استفاده برای این شی شامل color، border.width و border.color است که نام آن‌ها مشخص کننده‌ی عملکرد آن‌ها می‌باشد.

# TextArea:

یکی از component های مورد استفاده در این اپلیکیشن TextArea می‌باشد که در ادامه ویژگی‌های مورد استفاده در اپلیکیشن را توضیح می‌دهیم:

* id: همانطور که از اسمش مشخص است می‌توان به این شی یک آیدی را نسبت داد تا بتوان در بقیه‌ی اشیا از آن استفاده کردم.
* color: مشخص کننده رنگ متن می‌باشد.
* selectionColor: مشخص کننده‌ی رنگ هایلایت پشت متن انتخاب شده می‌باشد.
* selectedTextColor: مشخص کننده رنگ متن انتخاب شده می‌باشد.
* selectByMouse: مشخص می‌کند که آیا می‌توان این شی را با استفاده از موس انتخاب کرد یا نه.
* font.pixelSize: سایز فونت را با اندازه پیکسل مشخص می‌کند.
* background: رنگ پس‌زمینه را مشخص می‌کند.
* EnterKeyAction: در واقع این ویژگی مربوط به کیبورد مجازی می‌باشد که برای هر فیلد که ورودی می‌گیرد می‌توان مشخص کرد که دکمه‌ی enter چه عملکردی را داشته باشد. مقادیر مجاز آن به صورت زیر می‌باشد:
  + None: عملکردی ندارد
  + Go: با متن وارد شده به صورت آدرس برخورد می‌کند و به آن‌جا می‌رود
  + Search: عمل جست‌و‌جو را انجام می‌دهد.
  + Send: عمل ارسال را انجام می‌دهد.
  + Next: به ورودی بعدی می‌رود.
  + Done: گرفتن متن را به پایان می‌رساند و از کیبورد خارج می‌شود.

\* برای شی‌ها می‌توان ورودی تعریف کرد به عنوان مثال برای این شی دارای قطعه کد property int enterKeyAction: می‌باشد که با استفاده از property، ورودی enterKeyAction را تعریف کرده است. هر زمان از این شی استفاده شود می‌توان این ورودی را نیز به آن پاس داد.

# TextField:

ویژگی‌های مورد استفاده برای این شی مانند TextArea می‌باشد.

س

**منابع:**

<https://doc.qt.io/qt-5/qtvirtualkeyboard-index.html>

<https://resources.qt.io/videos/qtws17-using-virtual-keyboards-on-qt-embedded-devices-jan-arne-petersen-kdab>

<https://www.youtube.com/watch?v=JxyTkXLbcV4&t=74s>

<https://www.youtube.com/watch?v=ilCekEOpNYM>

<https://doc.qt.io/qt-5/qtqml-index.html>

<https://www.youtube.com/watch?v=GkzncJ71mm0>

<https://doc.qt.io/qt-5/qml-qtquick-virtualkeyboard-enterkeyaction.html>

<https://doc.qt.io/qt-5/qml-qtquick-controls2-textarea-members.html>