**لایبرری**

لایبرری مجموعه ای از کدهای از پیش کامپایل شده است که می تواند توسط برنامه ها دوباره استفاده شود و به دو دسته static و dynamic تقسیم می شوند

* **لایبرری static:** یک لایبرری استاتیک حاوی کدهایی است که در زمان کامپایل به برنامه های کاربران مرتبط می شود. فایل اجرایی تولید شده کپی خود را از کد لایبرری نگه می دارد.
* **لایبرری dynamic:** یک لایبرری داینامیک حاوی کدهایی است که برای چندین برنامه اشتراک گذاری شده است. محتوای لایبرری در زمان اجرا در حافظه بارگذاری می شود.

**پلاگین**

پلاگین یک library برای گسترش برنامه است که در run\_time اجرا می شود. مشخصه اصلی یک پلاگین کلاس ابسترکتی به اسم interface است که تمام متد های آن از نوع pure virtual است که کلاس پلاگین از این interface به ارث می برد. در زمان اجرا برنامه QPluginLoader پلاگین را بارگذاری می کند.

در کلاس اینترفیس برای اینکه به query این امکان رو بدیم در run\_time یک interface را در پلاگین پیاده سازی کند از Q\_DECLARE\_INTERFACE استفاده می کنیم.

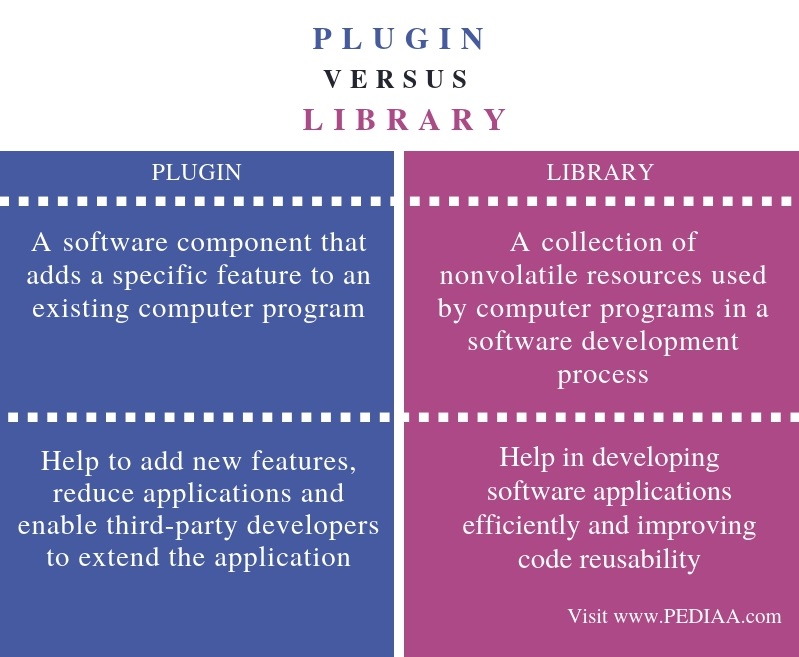
**Q\_DECLARE\_INTERFACE**

ماکرو ()Q\_DECLARE\_INTERFACE در interface دو ورودی میگیرد اولی اسم interface و دومی یک استرینگی است که اینترفیس را به روش منحصر به فردی شناسایی میکند (Java package name" syntax)

**نکته:** اگر بعداً اینترفیس ها را تغییر دهیم، باید از استرینگ دیگری برای شناسایی اینترفیس جدید استفاده کنیم. در غیر این صورت، برنامه ممکن است از کار بیفتد. بنابراین ایده خوبی است که ورژن را در استرینگ اضافه کنید.

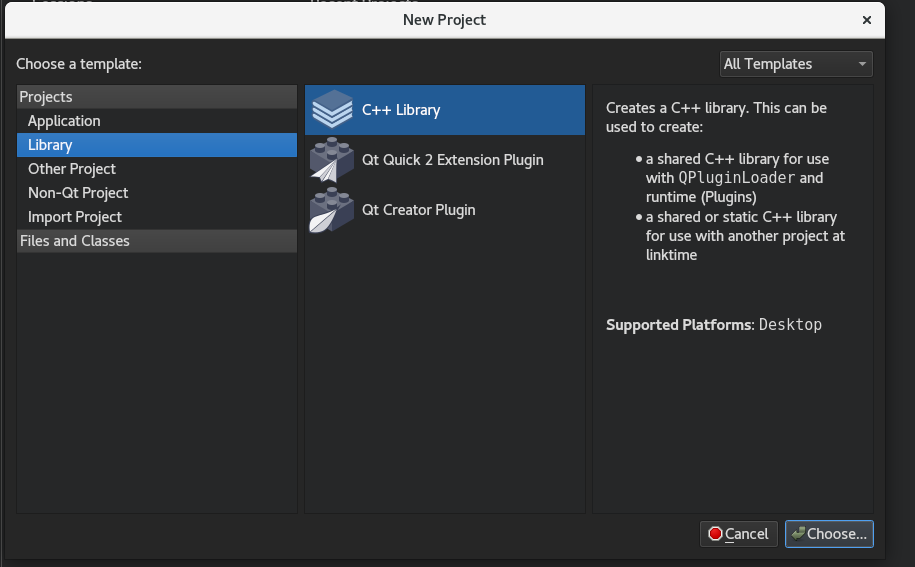
**کاربرد پلاگین**

ما برای گسترش یک برنامه و افزودن ابزار های مختلف به آن از پلاگین استفاده می کنیم که یک بار آن را نوشته و آن را در برنامه هایی که اینترفیس منحصر به فرد پلاگین را دارند استفاده می کنیم



**نحوه ایجاد لایبرری**

برای ساخت لایبرری ابتدا وارد new project شده در قسمت لایبرری مطابق شکل زیر ++C لایبرری را انتخاب می کنیم



سپس در قسمت type نوع لایبرری را انتخاب می کنیم (static/dynamic) .