

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

- **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 دینامیک حاوی کدهایی است که برای چندین برنامه اشتراک گذاری شده است. محتوای لایبرری در زمان اجرا در حافظه بارگذاری می شود.

پلاگین چیست ؟ پلاگین یک dynamic library برای گسترش برنامه است که در load run_time میشود. مشخصه اصلی یک پلاگین کلاس ابسترکته به اسم interface است که تمام متد های آن از نوع pure virtual است

کاربرد پلاگین : ما برای گسترش یک برنامه و افزودن ابزار های مختلف به آن از پلاگین استفاده میکنیم که بیار آن را نوشته و آن را در برنامه های مختلف استفاده میکنیم

مقایسه پلاگین و لایبرری : qt plugin ها در shared library ذخیره میشوند (DLL) که مزیتی نسبت به Qlibrary هایی که در shared library ذخیره میشوند دارند :

۱: کیوتی plugin loader بررسی میکند که یک پلاگین با ورژن کیوتی برنامه ما مرتبط است یا نه

۲: به جای اینکه شمارو مجبور کنه که یک c function را به صورت دستی resolve کنید , دسترسی مستقیم به یک ابجکت component ریشه (instance()) را فراهم میکنه

نکات کلاس اینتر فیس

برای اینکه به query این امکان رو بدیم در run_time یک interface را در پلاگین پیاده سازی کند از Q_DECLARE_INTERFACE استفاده میکنیم

ماکرو Q_DECLARE_INTERFACE() در interface دو ورودی میگیرد اولی اسم interface و دومی یک استرینگی است که اینتر فیس را به روش منحصر به فردی شناسایی میکند

(Java package name" syntax)

نکته: اگر بعداً اینترفیس ها را تغییر دهیم، باید از استرینک 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.

نکات کلاس پلاگین

در کلاس پلاگین، ماکرو Q_INTERFACE ضروریه به moc بگه که کلاس های base اینترفیس های پلاگین هستند

ماک (moc: (meta_object_compiler

ابزار ماک فایل های هدر ++c را میخواند اگر یک یا چند class declartion را پیدا کند که حاوی Q_Object macro باشد

یک فایل سورس ++c حاوی کد meta_object برای آن کلاس ها تولید میکند همچنین کد متا آبجکت نیاز به مکانیسم run_time type info , signal_slot و ویژگی های dynamic سیستم را دارد

فایل های سورس ++c که توسط moc ساخته شدند باید کامپایل و به implementation of the class وصل بشن

PLUG IN VERSUS LIBRARY

PLUG IN

A software component that adds a specific feature to an existing computer program

Help to add new features, reduce applications and enable third-party developers to extend the application

LIBRARY

A collection of nonvolatile resources used by computer programs in a software development process

Help in developing software applications efficiently and improving code reusability

Visit www.PEDIAA.com