هدف دادن دسترسی به node بدون دادن پوینتری به آن است :

در مثال جلسه قبل که پیمایش node هارا به خود لیست سپردیم اشکالی به می آید بدین گونه که اگر یک لیست از دو بخش بخواهد پیمایش بشود یا از طریق نفر به آن دسترسی گرفته شود , نقطه پیمایش هر کدام را نمی توان به صورت جدا نگه داشت

راه حل این موضوع تعریف کلاسی برای پیمایش کردن این لیست است که هر لیست میتواند نامحدود کلاس چیمایش کننده داشته باشد

بهتر است است هنگام پیاده سازی کلاس متد ها را به دو بخش command و query تقسیم کرد

که در این مفهوم command باعث تغییر آن object می شود اما query فقط اطلاعاتی را می خواند (Read Only)

چون کاربر در کلاس لیست نیازی به دسترسی مستقیم به node ندارد آن را در بخش Private کلاس لیست تعریف میکنیم (خود کلاس را)

نکته ای که در کلاس های تودرتو باید به آن توجه کرد این است اگر کلاس a داخل بخش private کلاس b تعریف شده باشد

بخش های عمومی کلاس a داخل کلاس b در دسترس اند

و بخش خصوصی اش در دستسرس نیستند

و اگر کلاس a در بخش public کلاس b تعریف شده باشد

بخش های عمومی اش در همه جا در دسترس اند

و بخش های خصوصی اش در دسترس نیستند

Template ها برای مشخص نکردن نوع داده هنگام نوشتن کد در انواع کلاس ها من جمله ساختمان های داده استفاده می شود

اما هنگام compile باید این نوع داده برای کلاس ها و توابع مشخص شود (چرا که C++ تمام template ها را در زمان compile با آن نوع داده مشخص شده جایگزین می کند )