

بسم الله الرحمن الرحيم

آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

نمودار کلاس (class diagram)

مدرس : مهندس افشین رفوآ

نمودار کلاس (class diagram)

نمودار کلاس یک نمودار **static** یا ایستا می باشد. این دیاگرام دید ایستا (**static view**) از یک برنامه را ارائه می دهد. تنها مورد استفاده ی نمودار کلاس در نمایش گرافیکی، توصیف و مستندسازی جنبه های مختلف یک سیستم خلاصه نمی شود، بلکه کاربرد دیگری دارد و آن هم ساخت کدهای قابل اجرا یک نرم افزار کاربردی می باشد.

نمودار کلاس خصیصه ها (**attribute**) و عملیات (**operation**) یک کلاس و همچنین محدودیت ها (**constraint**) اعمال شده بر روی سیستم را توصیف می کند. نمودارهای کلاس به طور گسترده به منظور مدل سازی سیستم های شی گرا بکار گرفته می شوند، زیرا نمودار کلاس تنها دیاگرامی است که می تواند توسط زبان های شی گرا نگاشت (**map**) شود.

نمودار کلاس مجموعه ای از کلاس ها، **interface** ها، **association** ها، **collaboration** ها و محدودیت ها را به نمایش می گذارد. نمودار کلاس زیرمجموعه ی نمودار ساختاری (**structural diagram**) می باشد.

کاربرد نمودار کلاس

هدف از بکار بردن نمودار کلاس مدل سازی دید ایستا (**static view**) از یک برنامه ی کاربردی می باشد. یادآور می شویم که نمودارهای کلاس تنها دیاگرام هایی هستند که توسط زبان های شی گرا قابل نگاشت بوده و بنابراین به طور گسترده در زمان ساخت مورد استفاده قرار می گیرند.

نمودارهای UML همچون نمودار فعالیت (activity diagram)، نمودار توالی (sequence diagram) تنها قادر به ارائه ی جریان توالی (sequence flow) برنامه ی کاربردی می باشند، اما نمودار کلاس کمی تفاوت دارد و بر اساس همین تفاوت پرکاربردترین نمودار UML در جامعه ی کدنویسان تلقی می شود.

هدف از بکاربردن نمودار کلاس در زیر به صورت خلاصه شرح داده شده:

1. تحلیل و طراحی دید static یک برنامه ی کاربردی.
2. شرح وظایف سیستم.
3. پایه ی نمودارهای component و deployment.
4. مهندسی معکوس (reverse engineering) و رو به جلو (forward engineering).

نحوه ی ترسیم نمودار کلاس

نمودارهای کلاس محبوب ترین نمودارهای UML هستند که برای ساخت نرم افزارهای کاربردی مورد استفاده قرار می گیرند. از این حیث یادگیری روال ترسیم نمودار کلاس از اهمیت بسیار بالایی برخوردار می باشد. نمودارهای کلاس خاصیت های متعددی دارند که حین ایجاد و ترسیم نمودار می بایست مورد توجه قرار داد، اما در اینجا نمودار از دید سطح بالا در نظر گرفته می شود.

نمودار کلاس در اصل یک نمایش گرافیکی از دید static سیستم مورد نظر بوده و جنبه های مختلف برنامه ی کاربردی مورد نظر را به تصویر می کشد. بنابراین مجموعه ای از کلاس ها یک سیستم کل را تشکیل می دهند.

لازم است به هنگام ترسیم یک نمودار کلاس، نکات زیر را یادآور شوید:

1. اسم نمودار کلاس بایستی معنی دار بوده و جنبه ی مورد نظر سیستم را توصیف کند.
2. تمامی المان ها می بایست به ضمیمه ی روابط آن ها از پیش شناسایی و مشخص شوند.
3. مسئولیت تمامی کلاس ها (متغیرهای عضو/خصیصه و متدهای آن کلاس) بایستی به روشنی مشخص و تعریف شود.

4. حداقل تعداد خاصیت ها یا متغیرهای عضو (**property**) ویژه ی هر کلاس بایستی مشخص شود. از خاصیت های غیر ضروری بایستی اجتناب کرد زیرا در صورت وجود خاصیت های غیر لازم نمودار بی دلیل پیچیده می شود.

5. هر جایی که فکر می کنید لازم است باید از **note** ها برای توصیف جنبه هایی از نمودار استفاده کنید تا بدین وسیله در انتهای فرایند ترسیم، نمودار برای توسعه دهنده/کد نویس قابل فهم باشد.

6. سرانجام، پیش از آماده سازی و ارائه ی نسخه ی نهایی، نمودار باید بر روی یک ورقه ی سفید ترسیم شده و هر تعداد دفعه که لازم بود آن را تکرار کنید تا خروجی صحیح درآید.

نمودار زیر مثالی از **order system** یک برنامه ی کاربردی است؛ تنها یک جنبه ی خاص از کل یک برنامه را توصیف می کند.

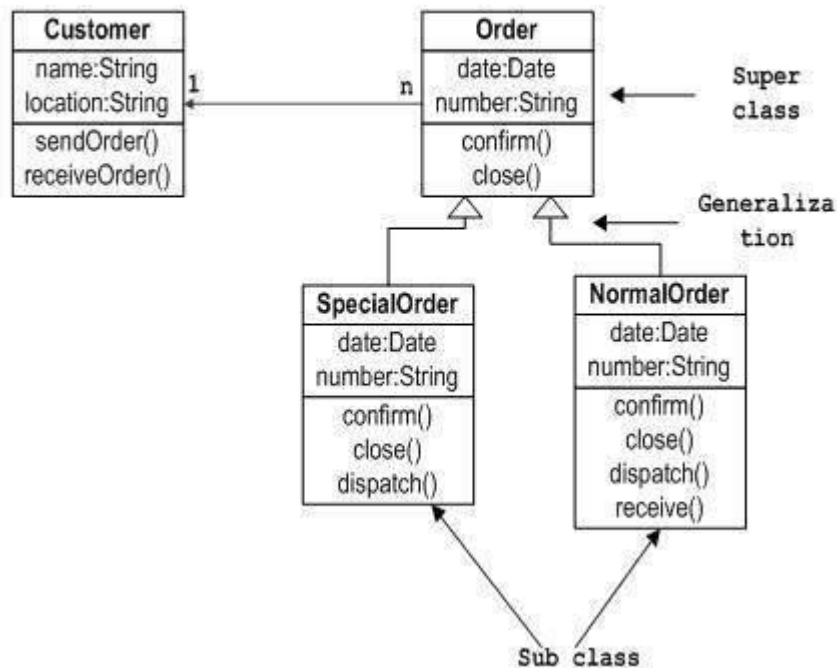
1. پیش از هر چیز، **Order** و **Customer** به عنوان دو المان اصلی سیستم مورد نظر شناسایی شده و این دو در یک رابطه ی یک به چند مشارکت دارند زیرا یک **customer** (مشتری) می تواند چندین **order** (سفارش) داشته باشد.

2. کلاس **Order** یک کلاس انتزاعی می باشد و دارای دو کلاس واقعی (از طریق رابطه ی وراثت) **NormalOrder** و **SpecialOrder** می باشد.

3. دو کلاس مشتق شده تمامی **property** های کلاس **Order** را به ارث برده اند. بعلاوه دارای توابع اضافی نظیر **dispatch()** و **receive()** هستند.

بنابراین نمودار کلاس با توجه به تمامی نکات ذکر شده در بالا ترسیم شده:

Sample Class Diagram



کجاها از نمودارهای کلاس استفاده می شود؟

باتوجه به آنچه قبلا شرح داده شد، نمودار کلاس یک دیاگرام **static** محسوب می شود و کاربرد آن در مدل سازی دید **static** یک سیستم می باشد. دید ایستا یا **static view** درواقع با استفاده از رابطه ها، اشیا، خصیصه ها و عملیات بر روی ساختار ایستا یک سیستم متمرکز شده و آن را تاکید می کند؛ بایستی گفت که دید ایستا **vocabulary** سیستم را توصیف می کند.

نمودار کلاس همچنین پایه و فونداسیون نمودارهای اجزا (**component**) و استقرار (**deployment**) محسوب می شود. نمودارهای کلاس تنها در نمایش گرافیکی دید ایستای سیستم کاربرد ندارد، بلکه این نوع نمودار در ایجاد کدهای قابل اجرا برای مهندسی معکوس و روبه جلو هر سیستمی مورد استفاده قرار می گیرد.

به طور معمول نمودارهای **UML** را نمی توان مستقیما توسط زبان های شی گرا نگاشت نمود، اما در این میان نمودار کلاس یک استثنا تلقی می شود.

نمودار کلاس نگاشت را توسط زبان های شی گرا همچون **Java**، **C++** و غیره .. پیاده سازی کرده و نمایش می دهد، از این رو با توجه تجربه ی عملی کسب شده می توان گفت که نمودار کلاس به طور معمول برای ساخت برنامه های کاربردی به وسیله ی زبان های شی گرا مورد استفاده قرار می گیرد.

موارد استفاده ی نمودار کلاس را می توان به صورت خلاصه نام برد:

1. توصیف دید ایستای سیستم.
2. نمایش همکاری بین المان های دید ایستا (ساختار ایستای سیستم را مورد تاکید قرار داده و نمودارهای ساختار ترکیبی، کلاس تحت پوشش آن قرار می گیرند).
3. شرح عملیاتی (**functionality**) که توسط سیستم قابل اجرا می باشد.
4. ساخت نرم افزارهای کاربردی توسط زبان های شی گرا.