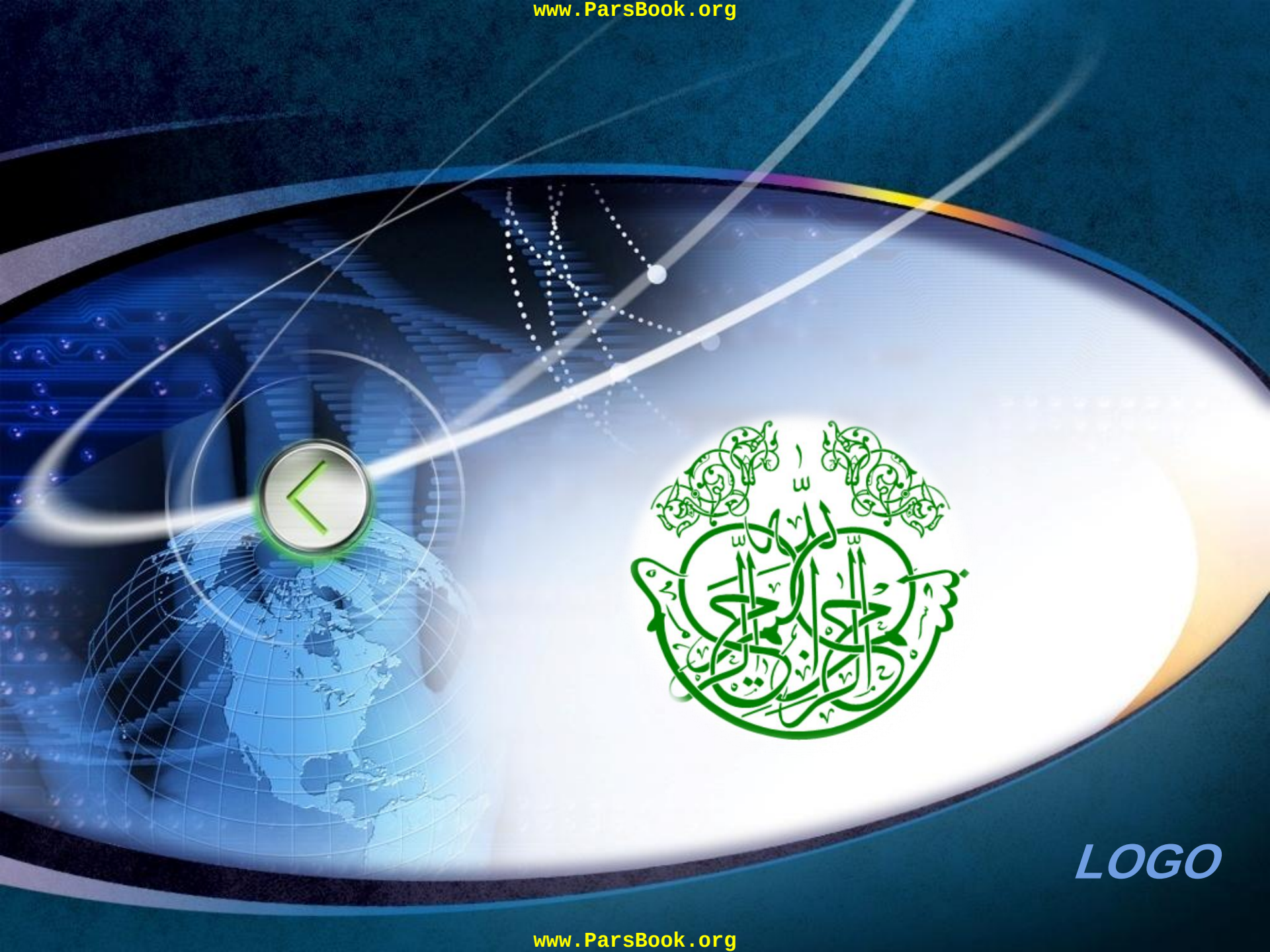


Object – Oriented

LOGO

. مقدمه

- شیءگرایی (Object – Oriented) لغتی است که امروزه در صنعت نرم افزار ، باب شده است . شرکت ها به سرعت حرکت می کنند تا خود را با این تکنولوژی جدید سازگار کنند و آن را در برنامه های موجود خود وارد نمایند . در حقیقت ، بیشتر برنامه ها امروز با شیءگرایی توسعه می بایند . اما شیءگرایی به چه معنا است ؟
- متد شیءگرایی یک راه متفاوت مشاهده برنامه هاست . با متد شیءگرایی ، شما یک برنامه را به قطعات بسیار کوچک یا آبجکت هایی تقسیم می کنید ، که تا اندازه ای مستقل از یکدیگر می باشند . به آن مانند ساختمانی از بلوک ها نگاه کنید .
- اولین قدم این است که تعدادی آبجکت های اساسی (انواع مختلف بلوک ها) را بسازید یا بدست آورید . اولین باری که شما این بلوک های ساختمانی را دارید ، می توانید آن ها را کنار هم گذاشته تا قصرتان را بسازید . به محض این که تعدادی آبجکت های اساسی را در دنیای کامپیوتر ساختید یا بدست آوردید ف می توانید به سادگی آن ها را کنار هم بگذارید تا برنامه های جدید را ایجاد نمایید . یکی از امتیازات اساسی متد شیءگرایی این است که می توانید یکبار Component (اجزاء) را ساخته و بارها و بارها از آن ها استفاده کنید ، درست مانند زمانی که می توانید یک بلاک ساختمانی را در یک قصر ، یک خانه یا یک سفینه فضایی دوباره استفاده کنید ، می توانید از یک قطعه طرح یا کد شیءگرایی در یک سیتم حسابداری ، یک سیستم بازرگانی یا یک سیستم پردازش سفارش استفاده مجدد نمایید .تفاوت متد شیءگرایی با روش سنتی توسعه ، چیست ؟

LOGO

- در روش سنتی ، روش توسعه به همراه اطلاعاتی که سیستم نگهداری خواهد کرد به خودمان وابسته است . در این روش ، ما از کاربران می پرسیم که چه اطلاعاتی را نیاز دارند ، پایگاه داده ای را طراحی می کنیم که اطلاعات را نگه دارد ، صفحاتی را تهیه می کنیم تا اطلاعات را بگیرد ، و گزارشاتی را چاپ می کنیم تا اطلاعات را برای کاربر نمایش دهد . به عبارت دیگر ، ما بر روی اطلاعات متمرکز می شویم و کمتر توجه می کنیم که چه کاری با این اطلاعات انجام شده است یا رفتار سیستم چگونه است . این روش data – centric (مبتنی بر داده) نامیده شده است و برای ایجاد هزاران سیستم در سال ، ایجاد شده است .مدلسازی data – centric مخصوص طراحی پایگاه داده و گرفتن اطلاعات خیلی مهم می باشد ، اما انتخاب این روش در زمان طراحی برنامه های تجاری با مشکلاتی همراه است . یک چالش بزرگ این است که درخواست های سیستم چندین بار تغییر خواهند کرد . سیستمی که از روش date – centric استفاده می نماید ، می تواند به آسانی تغییر در پایگاه داده را مدیریت نماید. اما اجرای تغییرات در قوانین تجاری یا رفتار (behavior) سیستم آن قدر آسان نمی باشد . متد شیءگرایی در پاسخ به این مشکل ، ایجاد شده است . با متد شیءگرایی هم بر اطلاعات و هم بر رفتار متمرکز می شویم . در نتیجه اکنون می توانیم سیستم هایی را ایجاد نماییم که انعطاف پذیر شده اند تا اطلاعات یا رفتار را تغییر دهند .
- مزیت این انعطاف پذیری با طراحی یک سیستم شیءگرایی به خوبی شناخته شده است . این مطلب ، به شناخت تعدادی از اصول شیءگرایی نیاز دارد . در اینجا شما می توانید ، با تعدادی از این اصول آشنا شوید .

LOGO

مزایای شیء گرایی :

۱- مدل سازی واقعی (مدل سازی مبتنی بر واقعیت) : دنیای واقعی ما نیز مجموعه ای از اشیاء است که با یکدیگر در تعاملند ، لذا روش شیء گرایی دقیق تر و صحیح تر این دنیا (جهان واقع) را مدل می کند .

۲- Reusability(استفاده مجدد) : در صنعت نرم افزار مقدار زیادی از زمان و انرژی صرف می شود تا کارهای تکراری انجام شود . در روش شیء گرایی شما کلاسهایی می سازید که می تواند در برنامه های مختلف بکار برده شود و این باعث صرفه جویی در زمان ، کوشش و در نتیجه هزینه می شود .

۳- resilience to change(قابلیت تغییر پذیری یا راحتی تغییرات) : در روش شیء گرایی برنامه ها می تواند به مرور زمان تکامل پیدا کند . در حقیقت درهنگامی که نیاز است تغییرات ایجاد شود نیازی نیست که همه برنامه از ابتدا نوشته شود ، بلکه با ایجاد کلاس جدید و ارث بری از کلاسهای قبلی می توانیم تغییرات را اعمال کنیم. همچنین اعمال تغییرات اثرات جانبی کمتری به وجود می آورد .

۴- ساختار ماژول ها در حالت شیء گرایی فاقد پیوستگی و دارای انسجام مناسب (بالاتری)است .

۵- نگهداری نرم افزارهای شیء گرایی ساده تر است .

LOGO

مفهوم سیستم :

1. تعریف سیستم
2. اجزاء سیستم (ورودی- خروجی- پردازش-کنترل- بازخور محیط سیستم)
3. انواع سیستم
4. خصوصیات سیستم باز:

holism کلی گرایی
interaction تعامل
purpose هدف
order نظم

inter dependence وابستگی اجزاء
hierarchy of system سلسله مراتب سیستمها
adaptive mechanism مکانیزم سازش
equifinality همپایانی

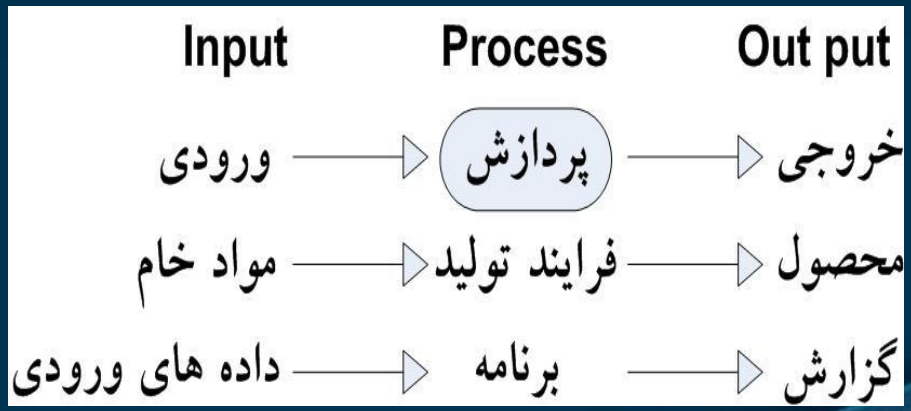
LOGO

سیستم :

مجموعه ای منظم از عناصر به هم وابسته است که برای رسیدن به اهداف مشترک با هم در تعاملند.
Interaction انسان ، منظومه شمسی ، تلفن و ...

اجزاء سیستم :

عواملی است که موجودیت آن را تشکیل می دهد و در رسیدن به اهداف سیستم آن را یاری می کند.



LOGO

سیستم با دو جزء باز خور و کنترل کامل می شود به چنین سیستمی ، سیستم سایبرنتیکی گویند. باز خور اطلاعاتی در مورد عملکرد واقعی سیستم است و کنترل فرایند سنجش و مقایسه عملکرد واقعی سیستم با عملکرد از پیش تعیین شده (استاندارد) می باشد.

بازخور مثبت با استاندارد ها می خورد.

بازخور منفی با استاندارد ها نمی خورد.

زمانی که نتایج عملکرد همسوی استاندارد ها باشد (بازخور مثبت)

زمانی که تفاوت و انحراف خروجی نسبت به استاندارد ها دارد(بازخور منفی)

کنترل بازخور منفی :

خود سیستمی را تشکیل می دهد که دارای اجزاء زیر است:

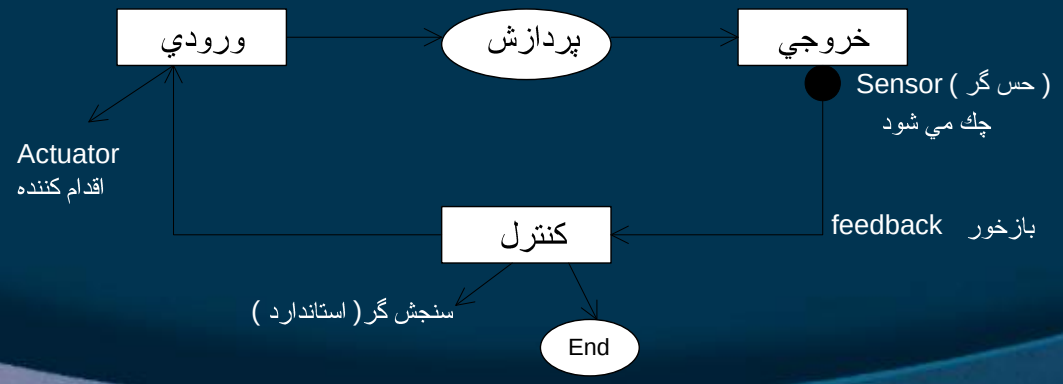
۱- ورودی بازخور

۲- واحد تست کننده **sencor** که واحد خروجی سیستم را سنجش می کند.

۳- واحد کنترل کننده که خروجی دریافت شده را با استانداردها مقایسه می کند.

۴- معیارها و استانداردهای مورد نیاز برای عملیات .

۵- واحد اقدام کننده **actuator** که اقدامات تصحیحی برای برطرف کردن انحرافات به ورودی سیستم ارسال می کند.



LOGO

محیط :

عامل خارج از سیستم که به رفتار سیستم اثر می گذارد و در واقع تعیین می کند که سیستم چگونه باید انجام وظیفه کند. مثلاً محیط یک سیستم تجاری شامل رقبا، خریداران، فروشندگان، دولت، سهامداران و .....

محدوده :

مرزی است که سیستم را از محیط خود جدا می کند.

انواع سیستم :

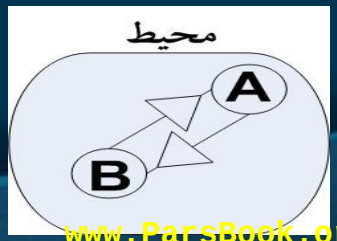
۱- سیستم های بسته ۲- سیستم های باز

سیستم بسته : محیطش هیچ گونه ارتباطی به صورت مبادله انرژی و اطلاعات ندارد مثل سیستم های فیزیکی و شیمیایی ،که طبق اصل دوم ترمو دینامیک :اجزای یک سیستم به طرف بی نظمی حرکت می کند ودر نهایت به حالت تعادل ایستا می رسد. Static equilibuium

سیستم باز: سیستمی است که به طور مستمر ورودی را از محیط دریافت کرده و پس از تبدیل به صورتی دیگر آن را به محیط باز می گرداند. مثل سیستم های بیولوژیکی اجتماعی و سیستمهای اطلاعاتی سازمانها و .../ سیستم باز از افزایش آنتروپی جلوگیری می کند و به تعامل پویا می رسد. Dynamic equilibuium

خصوصیات سیستم باز :

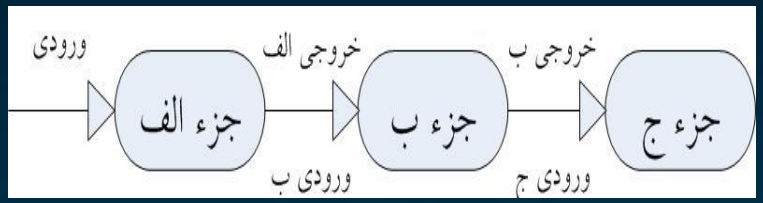
کل گرایی : منظور از کلمه مجموعه در تعریف سیستم جمع عناصر تشکیل دهنده آن نیست بلکه به یک کلیت و یکپارچگی اشاره می کند یک سیستم یک کل می باشد که فراتر از عناصر تشکیل دهنده آن است مثل آب که از اکسیژن و هیدروژن تشکیل شده و یا یک شرکت تولیدی که از قسمتهای مختلف مثل بازاریابی مالی و غیره تشکیل شده است.



LOGO

وابستگی اجزاء :

اجزاء سیستم به یکدیگر وابسته هستند یعنی ورودی یک جزء خروجی جزء دیگری می باشد این وابستگی و ارتباطات ادامه پیدا می کند تا سیستم به هدف خاص خود برسد مثلاً در یک واحد تولیدی سیستم خرید از سیستم تولید درخواستی برای خرید مواد می کند.



تعامل :

به مفهوم ارتباط متقابل سیستم ها با یکدیگر می باشد در حقیقت همه سیستم ها می توانند با دیگر سیستم ها رد و بدل اطلاعات داشته باشند.

هدف :

سیستم ها دارای هدف و مقصودی هستند و برای رسیدن به آن در تلاشند. مثلاً سیستم هواپیمایی هدفش حمل مسافر و بار به مقصد است. سیستم ها از نظر هدف به دو دسته تقسیم می شوند یا خود دارای هدف هستند مثل انسان یا اینکه هدف در آنها تعبیه شده است مثل یک ماشین که برای منظور خاصی طراحی شده است.

سلسله مراتب سیستم :

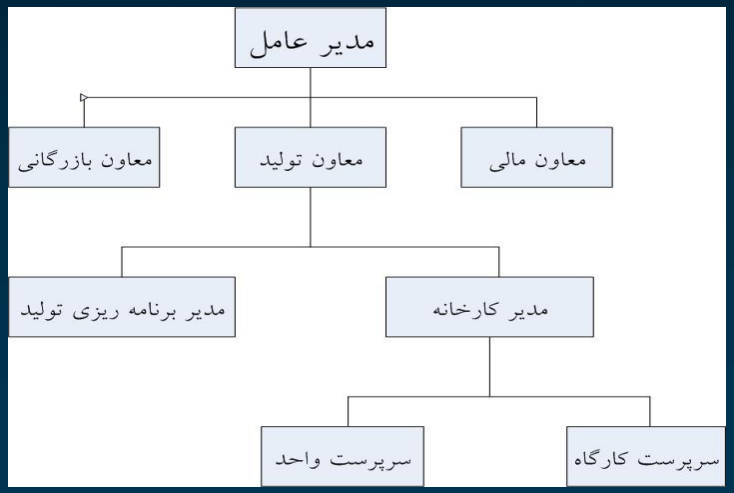
هرگاه عناصریک سیستم خود سیستم باشد یک زیرسیستم یا sub system نامیده می شود.هریک اززیرسیستم ها خود می توانند به زیرسیستم های دیگری تفکیک گردند .

LOGO

نظم :

حکایت از سازمان organization و ساختار دارد. Strnctuion اجزاء یک سیستم به گونه ای مرتبط سازمان دهی شده است که بتواند بر اساس یک برنامه از پیش تعیین شده با هم کار کنند. هر جزء نقش خاصی را بر عهده دارد . لذا ساختار بیان کننده نقش هر جزء و سازمان، بیان کننده برنامه کاری هر جزء و نحوه ارتباط با دیگر اجزاء می باشد.

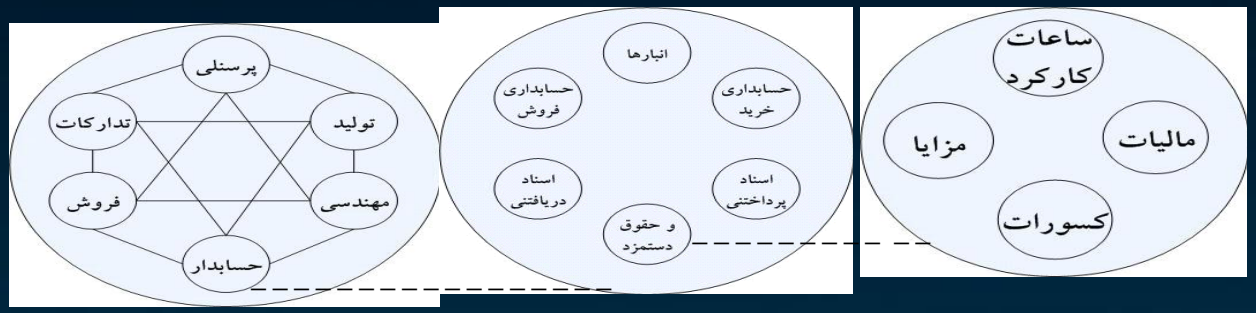
نمودار سازمانی :



در سازمانها نظم به صورت نمودار سازمانی می باشد. نمودار سازمانی به صورت سلسله مراتبی ارتباطات و مسئولیت ها را نشان می دهد. هر مستطیل نشان دهنده موقعیت افراد و خط از بالا به پایین نشان دهنده ساختار اختیارات جریان رسمی ارتباطات و دستورات را نشان می دهد .

LOGO

یک سیستم سلسله مراتبی :



شکل بالا از قسمتهای مختلف تشکیل شده و عکس این مفهوم نیز صادق است. موسسه تولیدی فوق می تواند زیر سیستمی از یک سیستم فراگیر supra system به نام صنایع تولیدی و صنایع تولیدی نیز بخشی از صنایع کل کشور باشد. این تفکیک پذیری سلسله مراتب سیستم ها را تشکیل می دهد.

مکانیزم سازش :

سیستم ها از یک طرف برای خود بایستی تغییرات محیطی را درک کرده و از تغییرات کلی که سیستم را به خطر می اندازد جلوگیری کند. از طرف دیگر سیستم باید تغییرات محیط خود را شناسایی کرده و خود را با آن تطبیق دهد. و بدین ترتیب با محیط به تعادلی پویا برسد. برای رسیدن به چنین تعادلی موسسات بایستی بازخور اطلاعاتی داشته باشد و در حقیقت باید از محیط چیز یاد بگیرد . در حقیقت این سیستم ها را سیستم های یاد گیرنده می گوییم.

همپایانی :

مفهومش این است که سیستم می تواند با ورودی های متفاوت به نتیجه نهایی یکسانی برسد. مثلاً یک موسسه یا سازمان با بالا یا پایین رفتن تقاضا به سود مورد نظر خود دسترسی پیدا کند.این نظریه بر خلاف نظریه علت و معلولی ساده است که از علوم فیزیک سرچشمه می گیرد و معتقد است که همیشه یک راه بهترین برای رسیدن به هدف وجود دارد.

LOGO

شیء :

هر چیز، مفهوم یا پدیده ای که قابل تشخیص باشد و دارای رفتار مشخصی از خود باشد یک شیء نامیده می شود.

هر شیء دارای ۳ مشخصه است :

State حالت- behavior رفتار- identity شناسه

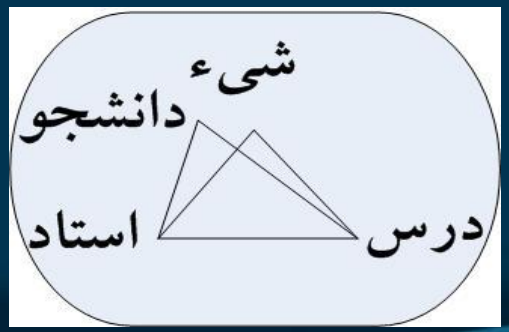
شیء ترکیبی از خواص و متدها است. مثلاً لامپ یک شیء است.

State : هر شیء خاصیتی دارد که بر اساس این خواص حالت شیء مشخص می شود . مثلاً حالت یک لامپ می تواند روشن یا خاموش باشد.

Behavior یا رفتار: تغییرات این خواص وحالتها در طول زمان می باشد. معادل تابع است. Method

State یا حالت : نشان دهنده مجموعه ای ازخواص(مجموعه ای ازمتغیرها) (attribute خاصیت)

شناسه : مثل شماره serial که روی هر لامپ است. چیزی است که هر شیء را به صورت منحصر به فردی مشخص می کند .



LOGO

کلاس :

مجموعه ای از اشیاء که ساختار و رفتار مشابهی دارند تشکیل یک کلاس می دهند . شیء و کلاس می توانند به جای هم بکار بروند.

ارتباط :

به ارتباط بین object های مختلف گفته می شود که در روش شیء گرایی این ارتباط از طریق فرستادن پیغام صورت می پذیرد. پیغام توسط شیء دریافت کننده ، تفسیر شده و بر اساس آن رفتاری از خود نشان می دهد.

Polimorphism (چند ریختی ، چند شکلی) :

اشیاء مختلف در مقابل یک پیغام مشابه ممکن است رفتار متفاوتی از خود نشان دهند به این تفاوت در پاسخ به یک پیغام ، Polimorphism یا چند شکلی گفته می شود.
بنابراین اگر شیء A تقاضایی از شیء B داشته باشد ، به آن یک پیغام می فرستد. کاری که شیء B برای شیء A باید انجام دهد توسط اجرای یک متد از شیء B صورت می گیرد . لذا اشیاء مختلف می توانند method های مختلفی برای پاسخ به یک تقاضای مشترک داشته باشند که این مفهوم چند شکلی است .

Abstraction :

(تجرید ، انتزاع) ، مفهوم Abstraction به این معنی است که شما فقط با جزئیات مورد نیازتان از یک شیء کار می کنید و آنها را می بینید . در حقیقت با این روش از پیچیدگی اشیا کم کرده و راحتتر با اشیاء برخورد می کنید.

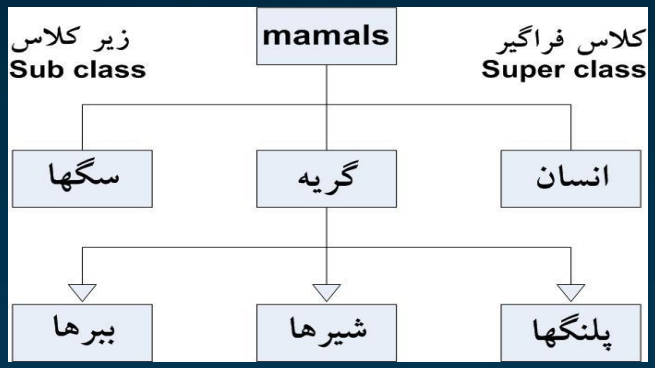
LOGO

کپسوله سازی :Encapsulation (data hiding)

مفهومی است که رابطه مستقیمی با Abstraction داردو به این معنا است که جزئیات غیرضروری را ازدسترس دورنگه دارید ، یا به عبارتی اطلاعات غیرضروری را مخفی کنید.

ارث بری و توارث : inheritance

کلاسهای مختلف می توانند در یک کلاس کلی تر تقسیم بندی شوند که این کلاس کلی تر شامل همه خواص مشترک کلاسهای زیرین می باشد در حقیقت می توان گفت کلاسهای زیرین از کلاس پدر (کلاس مشترک) خواص مشترک را به ارث می برند .



توارث ، راهی است که توسط آن مفهوم Abstraction اجرا می شود . زیرا هر چه قدر در این سلسله مراتب با کلاس های بالاتر کار کنیم. با مفهوم کلی تری کار می کنیم. و جزئیات غیر ضروری در زیر کلاسها را در نظر نمی گیرم در حقیقت جزئیات غیر ضروری کلاسها از دید شما مخفی شده است و در حقیقت این همان مفهوم Abstraction می باشد.

LOGO

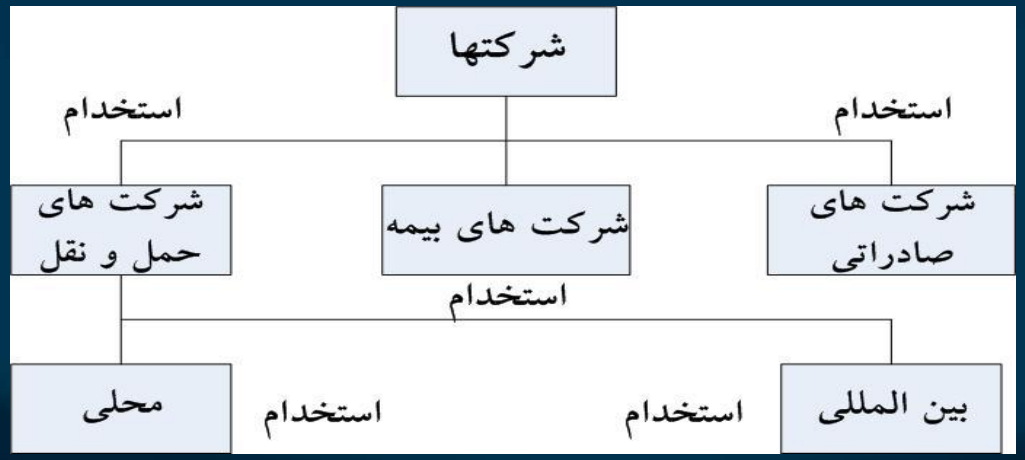
عمومی سازی :Generalization

خواص کلاسهای پایین را که مشترک هستند در کلاس دیگر قرار دهیم.
به سلسله مراتب کلاسها اگر از پایین به بالا نگاه شود مانند این است که خواص مشترک را از زیر کلاسها جمع کرده و در کلاس پایه جمع آوری کرده ایم . به این مفهوم Generalization می گویند.

اختصاصی سازی :Specialization

از بالا به پایین انتخاب می کنیم.(دسته بندی)

اگر به سلسله مراتب کلاسها از بالا به پایین نگاه شود مانند این است که شما کلاسهای خاصی ازکلاس عمومی مبنا را می سازید که این معنای Specialization می باشد.



LOGO

باز نویسی : Overriding

وقتی که یک کلاس که مجموعه ای از خواص و متدها می باشد ، ازکلاس فراگیر خود یا super class ارث بری کند ، یک خاصیت یا متد را با خاصیت و متد خود جایگزین کند گوییم که این خاصیت و متد بازنویسی یا overriding شده است .

ارث بری چند گانه :

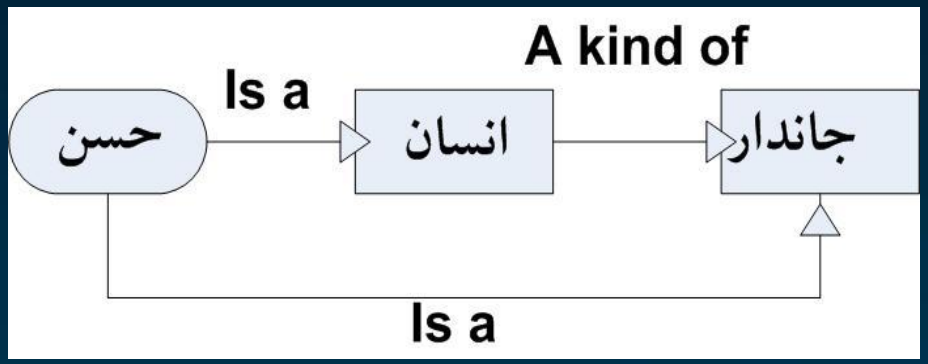
وقتی که یک کلاس خواص و رفتار خود را از دو یا تعداد بیشتری از کلاسها به ارث می برد گفته می شود که ارث بری چند گانه داشته است .multiple inheritance
ارث بری چند گانه باعث پیچیدگی های زیادی در برنامه نویسی می شود . زیرا خواص و متدهای مشترک در کلاسهای بالایی می تواند باعث ابهام در این کلاس گردد به همین دلیل زبانهایی مثل java و سی شارپ ، این نوع ارث بری را حذف کرده اند.

LOGO

ارتباط کلاسهای مختلف با یکدیگر : Relationship

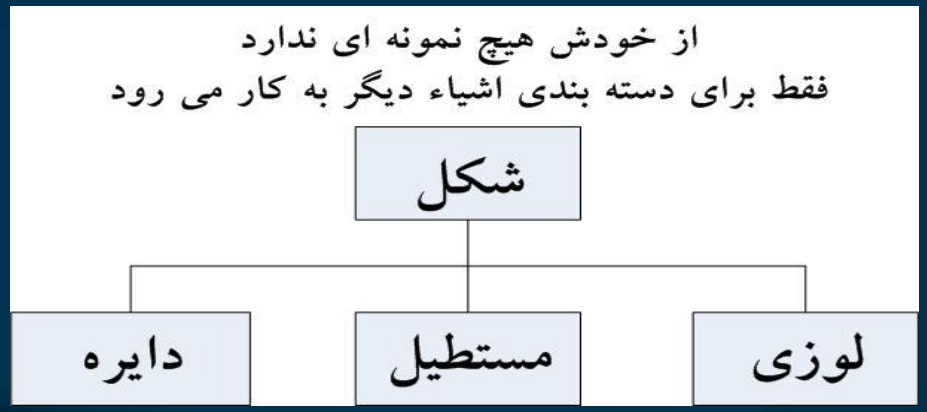
مثال : ارتباط بین انسان و جاندار
(ارتباط کلاس و کلاس)

(ارتباط شیء و کلاس)



کلاس مجرد : Abstract class

کلاسی که از آن نتوانیم هیچ object بسازیم.



LOGO

Interface (رابط) :

گاهی اوقات نیاز داریم که یکسری متد یا یکسری رفتارها را استاندارد کرده و این رفتارهای استاندارد را توسط کلاسهای مختلف پیاده سازی کنیم . مثلاً تلویزیون و ویدئو هردو به روش مشابهی نورتصویرراکنترل می کنند .منتهی تلویزیون به روش خود و ویدئو به روش دیگری این کار را انجام می دهند . ما می توانیم interface که شامل کم وزیاد کردن نوربه صورت کلی می باشد را تعریف کرده ، سپس کلاسهای مختلف مثل تلویزیون و ویدئو به روش خود زیاد کردن وکم کردن نوررا پیاده سازی نمایند .
در کلاس مجرد کلاسهایی که زیر کلاس این کلاس است و از آن ارث بری می کنند ، نوعی از همان کلاس مجرد می باشد مثل شکل، دایره، مستطیل و ....
در صورتی که interface مجموعه ای از رفتارها است که استاندارد شده و هر کلاسی که از آن ارث بری می کند آن را به نحوه خاص خود پیاده سازی می نماید.

اعضای استاتیک کلاس :

دانشجو

نام

نام خانوادگی

درس گرفتن

اعضای متغیر

اعضای ثابت کلاسها : هر کلاسی تشکیل شده از یک تعداد خاصیت یا attribute و متد ،که method ها و attribute ها به ازای هر object مقداری متفاوت دارند . به این اعضاء ، اعضای متغیر می گویند. در کنار این اعضا ما متدها و attribute هایی داریم که به ازای هر object مقدار متفاوت ندارد . در حقیقت این اعضا باید در سطح کلاس تعریف شوند ، که به آنها اعضای ثابت یا static می گویند .

LOGO

منابع :

۱- خود آموز UML در ۲۴ ساعت . ترجمه بشیری، الهام . انتشارات آموختگان – ناشر همکار الماس دانش .

۲- مرجع کامل UML with Rational Rose . مترجمان : مهندس توانا ، مهرداد و مهندس شیجونی ، عاطفه. انتشارات مؤسسه فرهنگی هنری نقش سیمرغ با همکاری گروه مهندسی - پژوهشی ساحر .

۳- اینترنت . (www.uml.ir)....

LOGO

تهیه کننده : صادق خیراللهی

Sadegh.kheyrollahi@gmail.com

LOGO

پایان

LOGO

SoftWare Engineer