

توسعه نرم افزاری با برنامه نویسی در ArcGIS

رحمت اله کاردان / عضو هیات علمی پژوهشکده سوانح طبیعی ایران
رسول حبیب زاده شجاعی / عضو هیات علمی دانشگاه آزاد اسلامی

چکیده

مدیریت منابع طبیعی، مدیریت و برنامه ریزی شهری به راحتی قابلیت تبدیل شدن به نقشه را دارند و در این حالت تحلیل اطلاعات و تصمیم گیری در برنامه ریزی ها را ساده تر و اصولی می نمایند.

رکن نرم افزار در سیستم اطلاعات جغرافیایی بصورت چند منظوره یا چندکاره (MultiTask) طراحی می شوند و با این خصوصیت دارای کاربردهای متنوعی هستند. اما حالت‌های ویژه وجود خواهند داشت که برخی از عملیات آن در نرم افزارهای GIS دیده نشده است و یا نیاز به تکرار برخی از فرامین و ایجاد توالی از آنها در یک عملیات ویژه GIS است که در این حالت، ماکرونویسی در داخل نرم افزار GIS تا ایجاد یک نرم افزار مستقل کاربردی با قابلیت های مبتنی بر GIS با استفاده از محیط برنامه نویسی ضروری می گردد. در زمان حاضر، ارائه سیستم اطلاعات جغرافیایی شیء‌گرا و کاربرد مفاهیم شیء‌گرای در طراحی و استفاده از این سیستمها رایج گردیده است. در اصول

با شروع قرن بیست و یکم، برخورداری از اطلاعات بهنگام و دارای کیفیت بالا برای حاکمان و مردم یک کشور، جنبه حیاتی و به مثابه خون زندگی بخش در رگهای موجود زنده است و جریان درست اطلاعات می تواند به شکوفایی گسترده کشور منتهی گردد. زیرا استفاده از اطلاعات در وجوه گوناگون برنامه ریزی کشور تاثیر بسزا دارد و شناخت صحیح ذخایر و منابع موجود کشور از نظر امکانات و استعدادها، نخستین گام در راه توسعه اقتصادی، فرهنگی و اجتماعی است و با برآورد تخمینی میزان ذخایر، منابع طبیعی و نیروی انسانی، امکان برنامه ریزی جامع در برنامه های کوتاه مدت تا بلندمدت میسر می گردد.

نرم افزارهای سیستم اطلاعات جغرافیایی (GIS) بر مبنای داده‌های نقشه‌ی و ارتباط آنها با داده‌های توصیفی، طراحی و به بازار رایانه‌ی دنیا عرضه شده‌اند و دارای ارکان سخت افزار، نرم افزار، داده‌ها و متخصصین خبره است. با GIS، بسیاری از اطلاعات زمینی جهت مطالعات زمین شناسی و معدنی، کاربری اراضی و

برنامه نویسی شیء‌گرا (Object Oriented Programming - OOP) مدل سازی، دنیای واقعی شیء‌ها با استفاده از نرم افزار ثانویه است، که اصل گرایی و ساده گرایی در مدل سازی، هدف اصلی در این شبیه سازی است. آشنائی هرچه بیشتر با شیء‌گرایی برای کاربران و کارشناسان متخصص در زمینه GIS فرصتی را فراهم می آورد تا بصورت شیء‌گرا در طراحی مدل از دنیای واقعی بیندیشند و برنامه نویسان این سیستمها با استناد بر اصول بنیادین شیء‌گرایی به توسعه نرم افزارهای خود برای ارائه به بازار اقدام نمایند و این مقاله بر مفاهیم برنامه نویسی شیء‌گرایی در محیط سیستم اطلاعات جغرافیایی و مبانی نظری ایجاد نرم افزار مبتنی بر قابلیت‌های مجموعه نرم افزاری ArcGIS تکیه خواهد داشت.

تعریف شیء‌گرایی

شیء‌گرایی (Object Oriented) یک روند طبیعی تفکر در مورد جهان و برنامه نویسی رایانه‌ای است. بطوری که انسان همیشه خواهان یافتن مرز بین پدیده‌های مختلف جهان هستی، برای تشخیص تا بکارگیری هریک از آن‌هاست. با نگاهی در جهان واقعی،

پیرامون هر مشاهده‌گری را شیء‌ها (Objects) فرا گرفته است. مردم، حیوانات، گیاهان، اتومبیل‌ها، ساختمان‌ها و رایانه‌ها همگی از این دیدگاه شیء محسوب می‌شوند و شیء‌گرایی ایده‌ی برگرفته از جهان واقعی است بطوری که برای یافتن راه حل مسائل انسانی باید نگاه ژرفی به جهان طبیعت انداخت.

این شیء‌ها از دیدگاه‌های مختلفی قابل تقسیم بندی هستند. بطور مثال با یک دیدگاه به دو دسته شیء‌های زنده و شیء‌های غیر زنده، که شیء‌های زنده در اطراف جابجا شده و کارهایی انجام می‌دهند. کلیه شیء‌های موجود در عالم هستی در برخی از خصوصیات با هم مشترکند. تمام شیء‌های دارای خصوصیات (Properties) همچون اندازه، شکل، رنگ و وزن هستند و رفتارها و روش‌های مخصوص به خود دارند، مانند بدن انسان که دارای روش‌های خاصی برای خوابیدن، راه رفتن، دویدن و غیره است و انسان‌ها از طریق مطالعه در صفات و رفتار خودشان به صفات و رفتار سایر شیء‌ها پی می‌برند.

مفاهیم شیء‌ها، خصوصیات، کلاس‌ها (Classes) ی‌کل و جزء

بهمراه اعضاء، از اصول اولیه تحلیل شیء‌گرایی است و انسان برای درک دنیای پیرامون خود، همواره از سه روش سازماندهی که بر کل افکار او حاکم است، استفاده می‌کند:

■ تفکیک تجربه خود به اشیای خاص و ویژگی‌های آن‌ها، مثال: وقتی انسان بین درخت و اندازه‌اش و روابط فضایی آن از دیگر اشیاء تمیز می‌دهد.

■ تمیز دادن بین شیء‌کل و دیگر اجزای آن - برای مثال: وقتی که انسان یک درخت و شاخ و برگ‌های آن را از یکدیگر تمیز می‌دهد.

■ شکل‌گیری و تفکیک طبقات مختلفی از اشیاء، برای مثال: وقتی انسان تمامی درخت‌ها را در یک کلاس (طبقه) و تمامی سنگ‌ها را در یک کلاس دیگر طبقه بندی و بین آن‌ها تمیز می‌دهد.

مفاهیم شیء و کلاس با تعریف ذیل بیان می‌گردد:

شیء انتزاعی از یک چیز در زمینه (یا حوزه) یک مسئله یا در زمینه پیاده سازی آن، که منعکس کننده قابلیت‌های یک سیستم برای نگهداری اطلاعات در باره آن، تراکنش با آن،

یا هر دو آن‌ها باشد، یک کپسول از ارزش‌های یک ویژگی و خدمات خاص آن.

کلاس: شرحی از یک یا چند شیء که بوسیله مجموعه‌ای از ویژگی‌ها و خدمات متحدالشکل توصیف می‌گردد، علاوه بر این، ممکن است که آن چگونگی ایجاد یا خلق اشیای جدید در یک کلاس را نیز بیان کند (یوردون، ۱۳۷۸).

خاصیت یک شیء نشاندهنده

رفتارها و ویژگی‌های شیء

در برنامه نویسی است.

رنگ هر عارضه یا

شیء خصوصیت

مشترک اکثر شیء‌ها

و کنترل‌ها در جهان

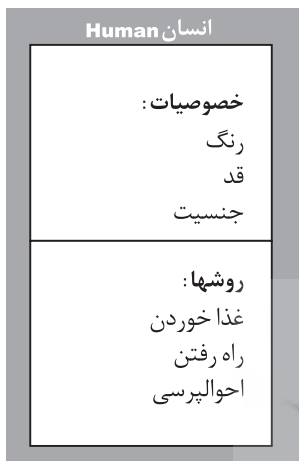
واقعی و جهان مجازی

برنامه نویسی است.

در حالی که برخی



کلاسهای مختلف را دارند بطوری که عوارض چندضلعی مرتبط با کاربری جنگلی می تواند بعنوان یک کلاس در سیستم شیء گرا معرفی گردد. زبان های شیء گرا، انتزاع را از طریق کلاس ها ارائه می دهند. کلاس ها، خصوصیات و روش های یک نوع شیء را تعریف می کنند و یک برنامه نویس از طریق هر کلاس، شیء مورد نظر خود را نمونه سازی (Instance) و ایجاد می کند.



محصورسازی

در محصورسازی، نمایش خصوصیات و روش های یک شیء بطوری انجام می گیرد که اجرای عملیات اصلی هر فرایند همانند جعبه سیاه از دنیای خارجی نهان گردد. بدین ترتیب موارد قابل اجراء نشان داده می شوند ولی نحوه انجام کار مخفی می ماند و امکان ایجاد شیء های مختلف به برنامه نویس داده می شود، در حالی که هیچ تاثیری برکدهای مورد استفاده نداشت. این کار با مفهوم واسط (Interface) هر شیء انجام خواهد شد.

چندریختی

اجرا های مختلف از شیء های گوناگون که یک عمل مشابه انجام می دهند، مفهوم چندریختی است. بطور مثال دنده برگشت (عقب) دو ماشین از دو نوع متفاوت می تواند یک عمل برگشتن ماشین را انجام دهد با وجود آنکه در ماشین نوع اول، دنده عقب بطرف جلو و ماشین نوع دوم، بطرف عقب حرکت می کند. چندریختی با استفاده از کلاس های متعددی که دارای واسط های یکسان هستند، کار خود را انجام می دهد. این فناوری اجازه ایجاد شیء را به کد برنامه نویسی شده می دهد و با استفاده از واسط ها، شکل دیگری از چندریختی را ارائه می دهد. چندریختی این امکان

خصوصیات خاص یک شیء هستند بطور مثال خصوصیت "مساحت" برای عارضه چندضلعی وجود دارد ولی برای عارضه نقطه ای یا خطی این ویژگی وجود ندارد. در برنامه نویسی شیء گرا، داده های توصیفی (Attributes) و توابع (رفتار) را در داخل بسته ای به نام شیء محصور می کند. داده ها و توابع موجود در یک شیء به صورت درونی به یکدیگر گره می خورند. شیء ها دارای خاصیت مخفی کردن اطلاعات هستند. این جمله به این معنی است که اگر چه شیء ها ممکن است بدانند که چگونه با دیگر شیء ها ارتباط برقرار کنند ولی شیء ها به طور عادی نمی توانند بفهمند که سایر شیء ها چگونه ایجاد شده اند و یا چگونه در درون آنها فعالیت انجام می شود و این جزئیات از شیء ها مخفی نگه داشته می شود. کلاس از روش ها، خصوصیات و رویدادها تشکیل شده است. خصوصیات کلاس انسان همانند رنگ پوست، قد و جنسیت است که تغییر آن ها بسیار دشوار (و یا غیر ممکن) است. روش ها، عمل هایی هستند که یک شیء، آن ها را اجراء می کند. غذا خوردن و احوالپرسی از روش های اعمالی برای کلاسی به نام انسان است که هر انسانی می تواند نوع غذا خوردن خود را به حالت های سنتی (آبگوشت یا چلوکباب) یا مدرن (پیتزا یا ساندویچ همبرگر) انتخاب نماید و راه شناخت و دستیابی به هر شیء با آشنائی با روشها، خصوصیات یا رویدادهای آن ممکن می گردد.

مفاهیم شیء گرای

توسعه شیء گرا، دارای چهار مفهوم کلیدی در مورد برنامه نویسی می باشد (فرانکین، ۱۳۸۰).
انتزاع (Abstraction)، محصورسازی (Encapsulation)، چندریختی (Polymorphism) و وراثت (Inheritance) است.

انتزاع

تصوری از شیء، به عنوان نوعی از اشیاء است که دارای برخی از روش ها (Methods) و خصوصیات ویژه است و می تواند برخی از اعمال را انجام دهد. بطور مثال، انسان (Human) می تواند شیء (کلاسی) در برنامه نویسی محسوب گردد که دارای برخی از خصوصیات رنگ، قد و جنسیت ویژه ای است و هر انسانی می تواند روش های مختلفی را برای غذا خوردن، راه رفتن و احوالپرسی داشته باشد. در شکل مقابل کلاس (شیء) "انسان" با سه خصوصیت و سه روش مختلف تصویر گردیده است. عوارض مکانی (گرافیکی) محیط GIS هم قابلیت تعریف در

را بوجود می‌آورد که یک برنامه نویسی بصورت گسترده به انواع متغیرهای موجود و کلاس‌ها مدنظر دسترسی داشته باشد. با استفاده از چندریختی طراحی اجزای سیستم بسیار آسانتر میشود. برنامه‌ها میتوانند به گونه‌ی نوشته شوند که پردازش عمومی تری بر روی ابرکلاس‌ها یا شیء‌های موجود سایر کلاس‌ها در سلسله مراتب انجام دهند.

وراثت

وراثت نظریه‌ی است که بر اساس آن یک کلاس می‌تواند بر مبنای کلاس دیگری به نام کلاس پایه ایجاد شود، وراثت، برای ایجاد سلسله‌هایی از شیء‌ها، یک ساز و کار (مکانیزم) ارائه می‌دهد. مثلاً سگ یک حیوان از خانواده پستانداران (Mammals) است و کولی (Collie) نوعی سگ می‌باشد. بنابراین، کلاس سگ (Dog) از خصوصیات و روش‌های خانواده پستانداران ارث می‌برد و کلاس سگ کولی از هر دو کلاس پستانداران (Mammals) و سگ (Dog) ارث خواهد برد.

تاریخچه کامپونت COM

در سال ۱۹۹۷ اولین شکل محیط اجرایی شیء‌ها از طریق شرکت میکروسافت ارائه گردید. هر یک از این شیء‌ها، یک برنامه کاربردی عموماً دارای فایل باینری است که نمی‌توان در آن تغییری ایجاد کرد، مگر اینکه نوع جدید آن را با نوع قبلی جایگزین کرد. کامپونت (COM) Microsofts Component Object Module در سطح باینری قرار دارد و بر این نکته دلالت دارد که آن‌ها می‌توانند در زبان‌های مختلف نوشته شوند و با هم در ارتباط باشند. کامپونت COM یک زبان برنامه نویسی است که توسط زبان‌های برنامه نویسی متفاوتی مانند ++C، پاسکال و سایر زبان‌ها می‌توان COM ایجاد کرد. انواع متفاوتی از واسطه‌های COM وجود دارند، تعدادی برای کنترل‌های اکتیواکس، تعدادی برای انتقال داده و برخی برای ذخیره‌سازی اطلاعات کاربرد دارند و بدین ترتیب، برای شیء‌های مختلف امکان تعامل با یکدیگر را فراهم می‌آورد. شرکت میکروسافت با توسعه کارایی COM در برنامه نویسی، سپس COM+ و COM توزیع شده (DCOM - Distributed COM) را به بازار ارائه نموده است. اکتیواکس (ActiveX Dynamic Link Library) نوعی از COM است که به خودی خود اجرایی نیستند و در یک نرم افزار دیگری مورد استفاده قرار می‌گیرند. هر برنامه کاربردی ویندوز که می‌خواهد اجرا شود بایستی در حافظه مقیم شود. DLL‌های

خصوصیات و روشهای شیء

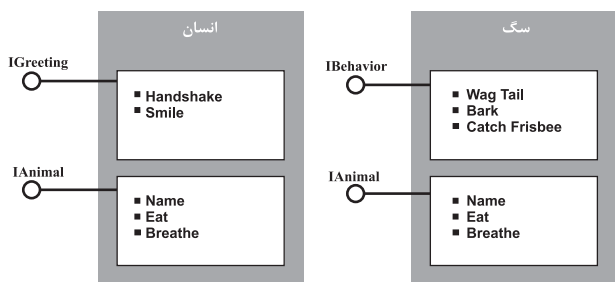
خصوصیات و روش‌های شیء بطور محصورشده در ارتباط با داده‌ها و روش‌های یک شیء دیگر به صورت درونی به یکدیگر گره می‌خورند و شیء‌ها دارای خصیصه پنهان کاری اطلاعات هستند. به این معنی که اگر چه شیء‌ها ممکن است با یک شیء دیگر ارتباط داشته باشند اما شیء‌ها اطلاعی از سایر شیء‌ها ندارند. جزئیات از دید خود شیء‌ها هم مخفی نگه داشته می‌شود. بطور مثال یک راننده را در نظر بگیریم که بدون اینکه از نحوه کار موتور و جزئیات آن اطلاعی داشته باشد، می‌تواند هدایت اتومبیل را در دست بگیرد. بچه‌های خردسال همیشه این سوال را دارند که «کودکان از کجا می‌آیند؟» که توضیح و جوابگویی به این سوال دشوارتر از آن است که گفته شود «شیء‌ها از کجا می‌آیند؟» در محیط شیء‌گرائی، شیء‌ها از کلاس‌ها تولید می‌شوند (Burke, ۲۰۰۲).

با تعیین شیء‌ها در برنامه نویسی که مرتبط با کلاسهای دارای خصوصیات و روشهای خاص هستند، ارتباط برقرار کردن و کدنویسی شیء با ترکیب کلی زیر انجام می‌گیرد که اول نام شیء و سپس روش یا خصوصیات آن شیء جهت انجام عملیات انتخاب می‌گردد:

Object.Method


بطور مثال، روش خوردن (Eat) یک سیب توسط شیء پسر (Boy) از کلاس انسان (Human) با ترکیب کلی بالا به صورت Boy.Eat (AnApple) خواهد بود. که پس از خوردن یک سیب توسط شیء پسر می‌توان خصوصیت گرسنه بودن (IsHungry) این شیء را از حالت بلی (Yes) به حالت نه (No) تغییر داد. این ترکیب کلی دستورنویسی در زبانهای برنامه نویسی شیء‌گرا بود و در ArcObjects با استفاده از مفهوم واسط خواهد بود.

واسط



شکل (۱) نمایش دو واسط برای کلاس انسان و دو واسط تعریف شده سگ

انسانها لحاظ نمود.

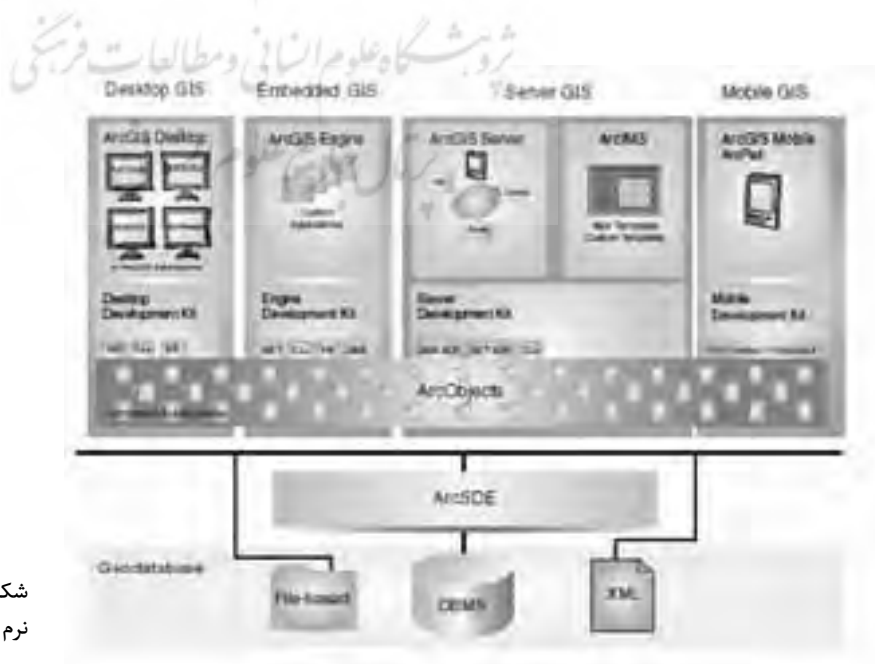
نامگذاری واسط ها بر اساس نام کلاس مورد نظر صورت می گیرد و نمادگذاری و مدلسازی نموداری آن از زبان مدلسازی UML^۲ استفاده گردیده است. حرف (I) جهت تمایز مابین کلاس ها و واسط ها قرارداد شده است و شکل گرافیکی یک واسط بصورت  در کنار شکل یک کلاس نمایش داده می شود.

سطوح برنامه نویسی در ArcGIS

برنامه نویسی در این نرم افزار یکی از روش های حرفه ی سفارشی کردن است که در اکثر نرم افزارها نسل جدید وجود دارد.^۲ سه روش کلی برای نوشتن کد و ایجاد نرم افزار کاربردی با زبان برنامه نویسی در ArcGIS با موتور ArcObjects وجود دارد:

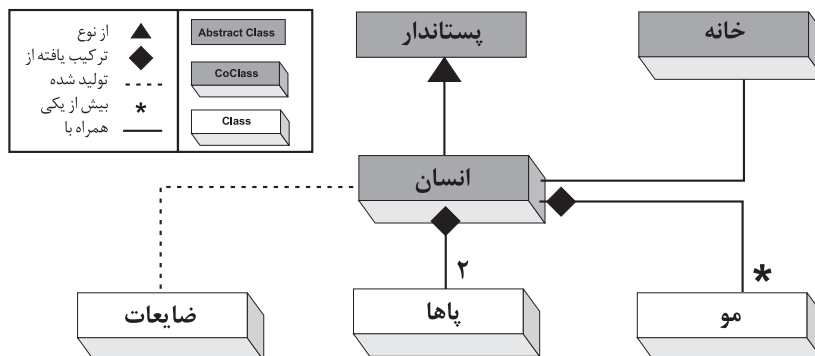
۱. محیط VBA و نوشتن کدها در داخل نرم افزارهای ArcGIS

واسط به گروهی از خصوصیات و روش های مرتبط باهم که دارای ارتباط منطقی هستند، گفته می شود و برای هر کلاس می توان چندین واسط در نظر گرفت. بسیاری از کلاس ها می توانند یک واسط یکسانی را اجرا کنند. متغیری که از یک کلاس نمونه سازی میشود، می تواند کلاس های اجرا کننده آن واسط را ارجاع دهد. بطور مثال برای خصوصیات و روش های غذا خوردن انسان ها، واسط IAnimal^۱ و برای نحوه رفتار انسان ها در احوالپرسی کردن، واسط IGreeting را در نظر گرفت. واسط ها انجام فرآیند توسعه کامل نرم افزار را راحت تر می کنند، بر تحلیل و طراحی تاکید دارند. واسط ها همانند لاستیک های یک گروه از ماشین ها هستند که بطور کلی برای چندین نوع ماشین طراحی و ارائه می گردند زیرا در دسترس بودن یک نوع لاستیک با قیمت ثابت و جنس مشخص در زمانی طولانی، کار بسیار دشواری است. برخی از واسط ها می توانند بطور مشترک مابین کلاس های مختلف تعریف شوند. در شکل (۱) واسط IAnimal برای کلاس انسان و سگ که شامل برخی خصوصیات منطقی مشترک بین انسان و سگ تعریف شده است و امکان طراحی واسط های دیگری برای هر یک از کلاس ها وجود دارد بطور مثال واسط ITalk در خصوص سخن گفتن انسان ها و طوطیان برای دو کلاس انسان و طوطی با توجه به تفاوت سخن گفتن این دو کلاس قابل طراحی است ولی باید در تعریف واسط ITalk برای طوطیان، خصوصیات و روشهای متفاوتی نسبت به واسط ITalk در



شکل (۲) جایگاه موتور ArcObjects در مجموعه نرم افزار ArcGIS

شکل (۳) نمودار مدل شیء خلاصه کلاس
پستاندار و ابرکلاس انسان



ارتباط سمبولها در نمودار مدل شیء

ارتباط مابین ابرکلاس ها، کلاس ها و خلاصه کلاسها همانند ارتباطات آنها در جهان واقعی است. برای سهولت یادگیری و نحوه ارتباط از نمودار مدل شیء (Object Model Diagram) استفاده می شود. نمودار مدل شیء جهت تسهیل در تعیین ارتباط مابین شیء ها و مستدل به قوانین زبان UML ایجاد می شود و هر شیء دارای نماد منحصر به خود است. بطور مثال، خلاصه کلاس بصورت یک مستطیل دویعدی و خاکستری رنگ دیده می شود و ابرکلاس بصورت سه بعدی و خاکستری رنگ است و از کلاس که بصورت سه بعدی ولی سفید رنگ است، تمایز داده می شود. شکل (۳) نمودار مدل شیء را در جهان واقعی و برای خلاصه-کلاس پستاندار و ابرکلاس انسان ترسیم کرده است. ارتباط هر سمبول با سمبول بعدی در گوشه سمت چپ - بالا شکل (۳) گویا شده است. بطور مثال اگر بیش از یک شیء بطور معلوم از یک کلاس در تشکیل کلاس بعدی استفاده شده باشد، تعداد آن در کنار خط ارتباطی یادداشت می گردد، مانند ۲ یا یک انسان و یا بیش از یکی و به تعداد نامعلوم در تشکیل کلاس بعدی شرکت داشته باشند از سمبول ستاره (*) استفاده می شود. ترکیب شدن کلاسها برای تولید یک ابرکلاس با لوزی و نوع کلاس شیء با مثلث نمایش داده می شود.

مراحل شناسایی یک شیء در ArcMap

با در نظر گرفتن ارتباطات گفته شده مابین سمبول های نمودار مدل شیء، می توان راه دستیابی به خصوصیات و روش های کلاس ها و ابرکلاس های مختلف را درک نمود و برای برنامه نویسی از آنها استفاده کرد. در شکل (۴) راه دستیابی به یک شیء (عارضه چندضلعی یک استان) از لایه گرافیکی چندضلعی استانها بصورت مصور نمایش داده شده است. در این شکل هر یک از ابرکلاس ها، خلاصه کلاس و کلاس عارضه (Feature) با

۲. استفاده از محیط ویژوال بیسیک، C، VB.NET و... که تولید اجزاء اکتیوآکس COM همانند DLL یا OCX گردد و با فراخوانی آن در نرم افزارهای ArcGIS این اجزاء بکارگرفته شوند.

۳. نوشتن یک برنامه اجرایی کامل و مستقل در محیط ویژوال بیسیک، ویژوال C یا VB.NET و... با اجزاء اکتیوآکسهای ArcObjects

سه روش فوق، به ترتیب از سطح توسعه داخلی نرم افزار تا پیشرفتهترین حالت توسعه نرم افزار در سیستم اطلاعات جغرافیایی است و با توانایی های توسعه گر ارتباط تنگاتنگ دارد و انتخاب سطح توسعه در گرو قدرت توسعه گر و توانایی اوست. جایگاه موتور ArcObjects در توسعه نرم افزاری مجموعه نرم افزار ArcGIS در شکل (۲) دیده می شود.

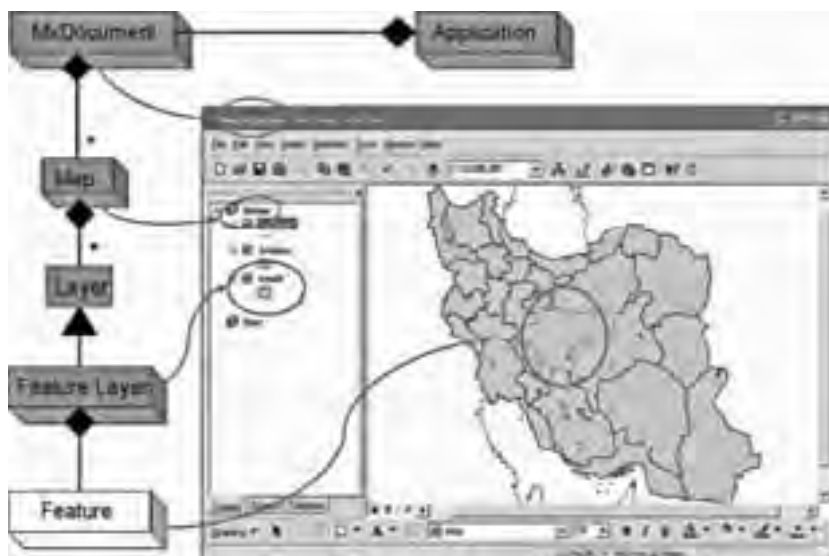
تقسیم بندی کلاس شیء ها

کلاس شیء های ArcObjects در سه دسته کلی زیر تقسیم بندی شده اند:

ابرکلاس (CoClass): کلاس شیء هایی که قابل نمونه برداری (Instantiable) و قابل تولید (Creatable) با کلمه واژه New را دارا هستند، در این طبقه قرار می گیرند.

کلاس (Class): کلاس شیء هایی که قابل نمونه برداری اند ولی غیرقابل تولید هستند و با کلمه واژه New تولید نمی شود بلکه به روش های دیگر تولید می شود.

خلاصه کلاس (Abstract class): کلاس شیء هایی که غیرقابل نمونه برداری و غیرقابل تولید هستند و تنها نگهدارنده واسط های تعریف شده در ابرکلاس ها و کلاس ها می باشد.



شکل (۴) مراحل شناسایی یک شیء (عارضه چندضلعی استان) در ArcMap

توجه به نمادگذاری مدل شیء آورده شده است. برای توسعه در نرم افزارهای کاربردی و ماکروها در ArcGIS از شیء های ArcObjects استفاده می گردد. بطور مثال در محیط VBA، برای رسیدن به شیء مورد نظر باید از شیء سراسری Application شروع کرد و به طور پلکانی به شیء مدنظر دسترسی پیدا نمود. شیء سراسری Application در تمامی توابع VBA قابل دسترسی است و نیازی به فراخوانی مرجع دادن آن نیست (Razavi, 2002).

سیستم شیء گرا دارای واسط است که واسطها از کلاسها ایجاد می شوند و هر

کاربردی (Visual Basic for Application - VBA) یکی از فن آوریهای توسعه ی شرکت میکروسافت در اکثر نرم افزارهای ویندوز همانند Word، Excel، Access از خانواده Office MicroSoft است. برخی از شرکتهای تولیدکننده نرم افزار ویندوز مانند AutoDesk، ESRI و... قابلیت توسعه نرم افزار خود را با محیط VBA امکان پذیر کرده اند و با این محیط، ماکروهای تهیه شده در داخل همان نرم افزار نوشته، فراخوانی و اجراء می شود.

قابلیت دسترسی آسان در اکثر محیط های نرم افزاری، اولین ویژگی مثال زدنی در زبان برنامه نویسی ویژوال بیسیک محسوب می شود. خواندن بخشی از کدنویسی در این زبان و درک اتفاقات موجود برای هر برنامه نویس این زبان، کار ساده و قابل فهمی است. ویژوال بیسیک در طی دوران عمر خود به یک ابزار توسعه شیء گرا تبدیل شده است و از قواعد زبان های شیء گرا و مباحث وراثت، محصورسازی و... در آن استفاده بسیار می گردد. از دیگر قابلیت های این زبان می توان به موارد زیر اشاره نمود:

- دارای دستوراتی شبیه به زبان محاوره ی و ایجاد برنامه های کاربردی و مفید با کمترین زمان ممکن در اکثر نرم افزارهای کاربردی در محیط ویندوز است.
- محیط دیداری مناسب و ارجاعات اجزاء سیستم و دیگر نرم افزارهای نصب شده در ویندوز برای کاربردی شدن محیط با کمترین برنامه نویسی
- دارای پنجره های کاربر پسند برای تهیه یا اجراء یک ماکرو

کلاس با یک واسط مرتبط با خود می تواند در خدمت کلاس دیگری باشد. یک ابرکلاس شامل ترکیبی از کلاس ها و روش های مورد نظر است. یک کلاس جز یک ابرکلاس است و مالکیت خود را از واسط ابرکلاس به دست می آورد. شیء را در ابرکلاس ها می توان با استفاده از کلمهواژه New ایجاد نمود. با اعمال کردن عملگر کلمه واژه New به یک ابرکلاس، یک نمونه شیء ایجاد می شود. خط کد زیر مثالی از ایجاد نمونه شیء pMap از ابرکلاس MxDocument با استفاده از واسط IMxDocument است.

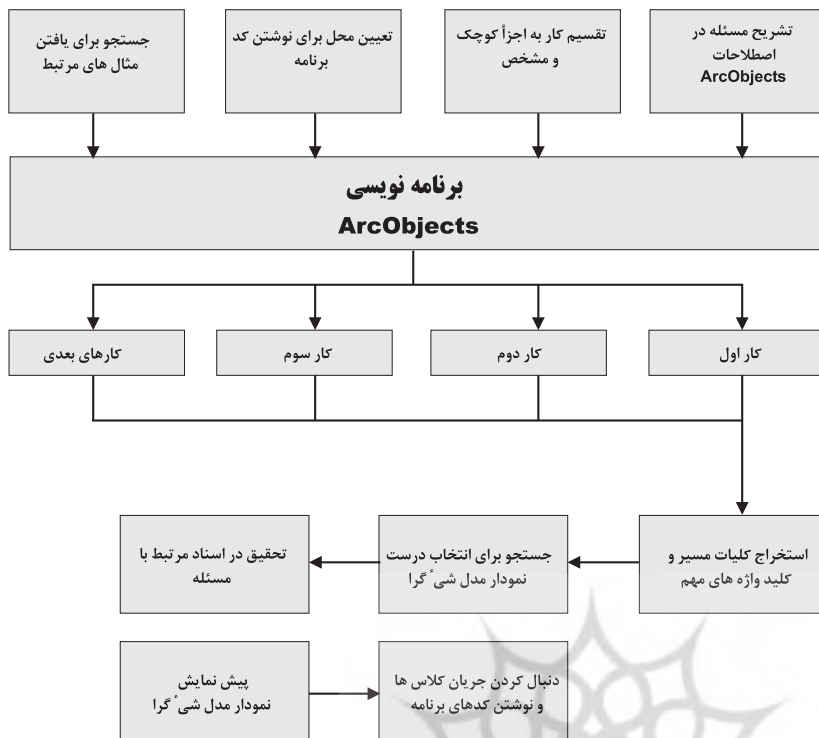
Dim pMap as New IMxDocument

حل مسئله با ArcObjects و تعریف کارهای قابل برنامه نویسی

اولین کار برای یک برنامه نویسی، تجزیه و تحلیل درستی از مسئله است که با درک صحیحی از مسئله آغاز می گردد. برنامه های رایانه ای جوابگوی مسائل دنیای واقعی هستند و مسائل واقعی نیاز به برنامه های بزرگ و پیچیده دارد، ولی برای حل مسائل باید شناخت مناسب از سلسله مراتب مسئله و درک درستی از آن داشت. تجزیه مسئله به قسمت های کوچکتر، که هر قسمت وظیفه خاصی را انجام دهد، شروع کار برنامه نویسی است و داشتن مدیریت و نحوه اجرای آن با الگوریتم و نمودار گردش اطلاعات است.

مروری بر محیط VBA

برنامه نویسی در محیط ویژوال بیسیک برای نرم افزارهای



شکل (۵) نمودار قرارگیری صحیح مدل شیء گرا و حل مسئله

■ ویژگی های برنامه شیء گرا و رویدادگرا^۲ را در خود ایجاد نموده است.

■ راهنمای سنجش هوشمند^۵ برای کامل کردن خصوصیات و روش ها در کدنویسی است. از ویژگی های این محیط حالت دیداری و سنجش هوشمند به کنترل آن همانند دیگر نرم افزارهای ویژوال است. بطوری که بعد از تایپ نام هر کنترل یا شیء تعریف شده در روال جاری و تایپ یک کاراکتر نقطه اعشار(.)، تمام خصوصیات و روش های آن کنترل یا شیء بصورت منوی کشویی باز شده و انتخاب از گزینه های آن میسر می گردد (شکل زیر).

ویرایشگر کد محیط VBA و داخل نرم افزارهای ArcMap، ArcCatalog و یا ArcScene قابل اجرا است و با داشتن تجربه در این محیط میتوان در محیط ویژوال بیسیک یا C و ... در ArcGIS به توسعه نرم افزارهای کاربردی پرداخت.

برای باز کردن محیط VBA با فعال کردن یکی از نرم افزارهای ArcMap یا ArcCatalog و انتخاب منوی Tools و گزینه Visual Basic Editor «Macros»، پنجره مربوطه باز میگردد، همچنین باز کردن این پنجره با کلید میانبر Alt+F11 همانند نرم افزار اتوکد امکانپذیر است. با باز شدن محیط برنامه نویسی VBA متعامل با نرم افزار فراخوان، چندین پنجره با اسامی مختلف همانند شکل (۶) ظاهر می گردند:

۱. پنجره پروژه (Project Window)


۲. پنجره خصوصیات (Properties Window)

۳. پنجره ویرایشگر کد (Code Editor)

در پنجره پروژه ابزار کاوشگر پروژه وجود دارد که با کلیک کردن روی کادرهای علائم مثبت (+) یا منفی (-) داخل کاوشگر پروژه، پروژه های شخصی قابل دسترسی می گردند. خصوصیات هر شیء یا کنترل انتخاب شده در پنجره خصوصیات نمایش داده



بدین سبب است که اکثر نرم افزارهای ویندوز پایه همانند اتوکد تا نرم افزارهای فن آوری GIS تولید شده از دیگر شرکت های رایانه ای، از این زبان برنامه نویسی بعنوان یکی از پایه های توسعه ی خود استفاده می کنند و تفاوت برنامه نویسی در هریک مربوط به اصطلاحات و عملیات رایج در همان نرم افزار است و از نظر کلیات با یکدیگر تفاوت چندانی ندارند. محیط VBA ویرایش ساده از ویژوال بیسیک در ArcGIS است^۶ که برای شروع برنامه نویسی در ArcGIS و ایجاد ماکروهای کاربردی در این محیط با موتور ArcObjects فراهم شده است. با نوشتن ماکرو های VBA در ArcGIS، تجربه برنامه نویسی بمنظور تهیه نرم افزارهای کاربردی بدست خواهد آمد و در این محیط تغییر کد برنامه ها با سرعت بالایی قابل انجام است. در محیط VBA نمی توان یک فایل اجرایی مستقل ساخت و همیشه این کدها در

میله منو محیط VBA همانند دیگر نرم افزارهای ویندوزی دارای منوهایی با وظایف خاص خود است و قابلیت سفارشی کردن دارد. تغییر نمایش پنجره نرم افزار ArcMap از محیط VBA و بلعکس آن از کلید میانبر Alt+F11 و خروج از محیط VBA با دکمه بستن نرم افزار و یا کلید میانبر Ctrl+Q امکانپذیر است. اجرای کد نوشته شده با منو و گزینه RUN یا کلید میانبر F5 در محیط VBA انجام می گردد و تغییر در کدنویسی بلافاصله با فشار دکمه F5 در بوته آزمایش و اجرا قرار می گیرد. در میله ابزار استاندارد محیط VBA با فشار دکمه  اجرای بعد از نوشتن کدهای مورد نظر، برنامه نویس شاهد عملکرد کدها خواهد بود.



شکل (۶) محیط برنامه نویسی VBA متعامل با نرم افزار ArcMap

ساماندهی کدها

در پنجره پروژه، سلسله مراتب ذخیره و ساماندهی کدها در کاوشگر پروژه اجرا می شود. هر پروژه می تواند چندین مدول داشته باشد که هر مدول تعدادی روال با خط-کدهای برنامه نویسی را با توالی زیر شامل می گردد (شکل ۷).

دستیابی به کدها به منظور اجرای یک روال یا ویرایش آن از طریق کاوشگر پروژه و پنجره ویرایشگر انجام می شود و تعیین محل ذخیره کدهای نوشته شده در پروژه به عهده برنامه نویس است و در سه حالت یا الگوی مختلف قابل مکان گزینی است. شکل (۸)، اصطلاحات روال، مدول، پروژه و کاوشگر پروژه در محیط VBA را در دو پروژه کلی با عنوان Normal.mxt و Project(KARAN.mxd) به تصویر کشیده است.

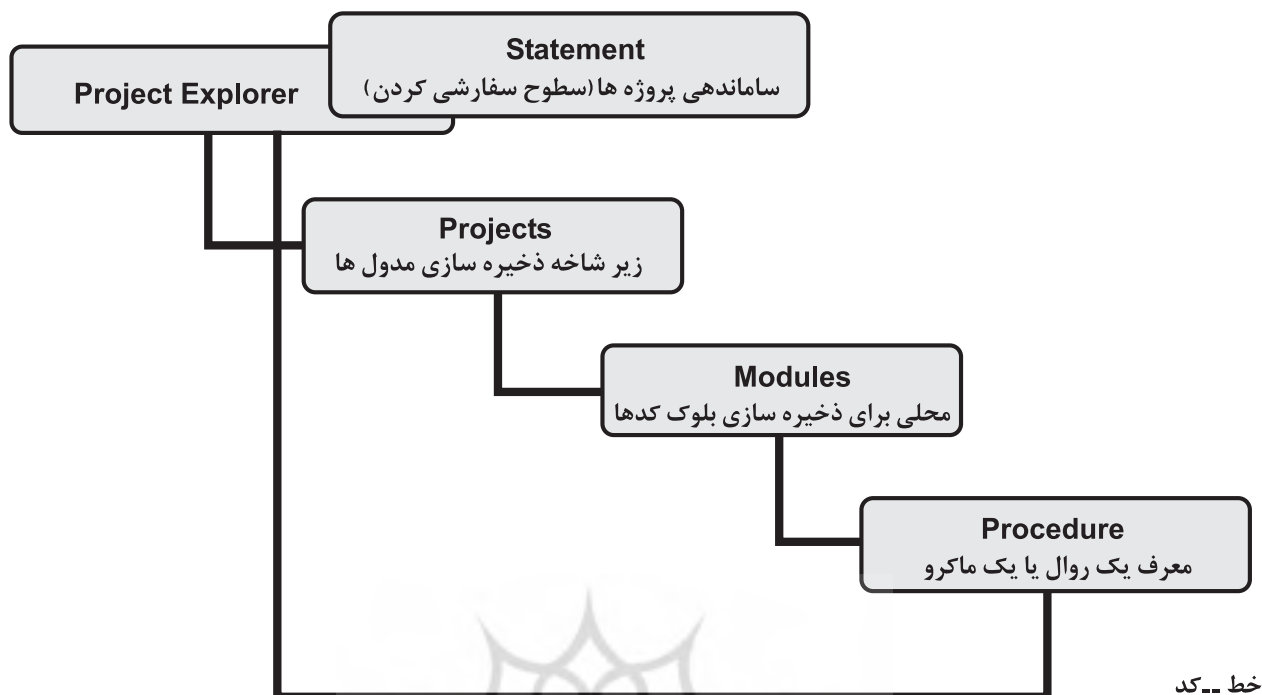
نتیجه

پیشرفت فن آوری GIS با توسعه نرم افزارهای قدرتمندی همراه بوده است که در علوم مرتبط کارایی خود را به ثبوت رسانده اند ولی در کارهای حرفه ای نیاز به توسعه نرم افزارهای کاربردی و سفارشی، اهمیت خاص دارد که این توانایی در حیطه برنامه نویسی در GIS امکانپذیر است. در این مقاله، مفاهیم اولیه برنامه نویسی شیء‌گرایی در سیستم اطلاعات جغرافیایی مبتنی بر موتور ArcObjects معرفی گردید و با مروری بر اصطلاحات رایج شیء‌گرایی همچون انتزاع، محصورسازی، چندریختی، وراثت و واسط به سه سطح برنامه نویسی در ArcGIS اشاره شد. تقسیم بندی کلاس شیء‌ها در برنامه نویسی محیط ArcGIS با عنوان ابرکلاس، کلاس و خلاصه کلاس به همراه یک

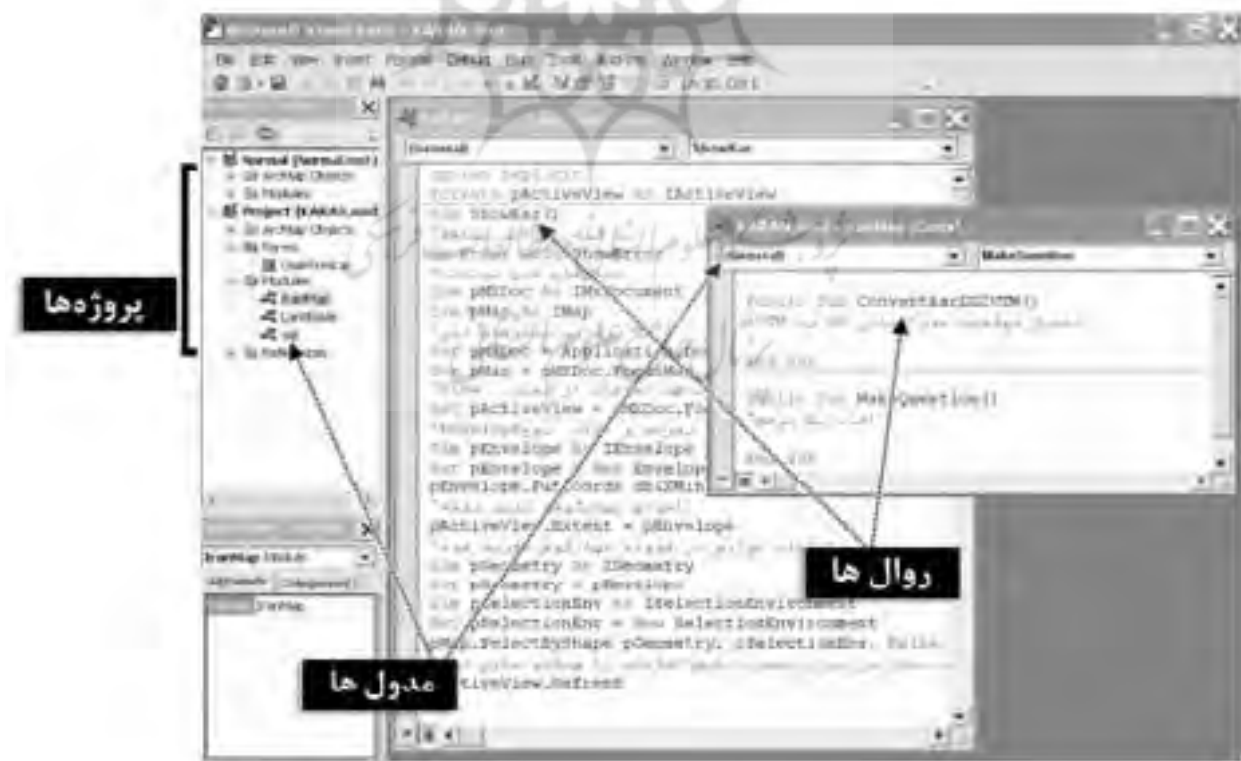
می شود و برخی از خصوصیات در همین پنجره قابل ویرایش هستند. امکان برنامه نویسی در پنجره ویرایشگر کد بوجود می آید که کاربرد هر یک توضیح داده خواهد شد.

میله منو (Menu Bar) در محیط VBA همانند دیگر برنامه های کاربردی ویندوز شامل فرمان های مرتبط با هم در منوها و وابسته به نوع کاربرد هایشان هستند. برخی از این کاربردها همانند فرمان ذخیره (Save) و بازکردن (Open) به کرات در یک اجرای برنامه قابل استفاده است، که این کار به روش منویی و گاهی اوقات بصورت دکمه های میانبر انجام می شود. برخی از فرامین در مواقع خاص استفاده می گردند، بطور مثال موارد زیر از آن جمله هستند:

- در منوی پرونده (File) گزینه Import برای کدهای نوشته شده یا فرمهای طراحی شده از قبل و گزینه Export برای ایجاد فایل خروجی یا پشتیبانی از کدهای نوشته شده فعلی استفاده می گردد.
- منوی ویرایش (Edit) امکان تغییرات و اصلاح بر روی عملیات برنامه نویسی را مهیا می سازد.
- در منوی درج (Insert) قابلیت افزودن یک فرم کاربردی، مدول یا روال وجود دارد.
- منوی اشکال زدائی (Debug) برای یافتن اشکالات احتمالی در برنامه نویسی و رفع آنها به طرق مختلف تعبیه شده است.
- منوی اجرا (Run) گزینه های اجرای برنامه و بصورت گام به گام را در خود دارد.
- منوی ابزارها (Tools) جهت ارجاعات و اختیارات امکان پذیر در این محیط مهیا گردیده است.



شکل (۷) نحوه ساماندهی کدهای یک پروژه



شکل (۸) روال، مدول، پروژه و کاوشگر پروژه در محیط VBA



مثال از مراحل شناسایی این تقسیم بندی در نرم افزار ArcMap انجام گردید و نمودار گردش اطلاعات برای حل یک مسئله با برنامه نویسی ارائه شد. در انتها، یکی از سطوح برنامه نویسی معرفی شده در محیط VBA با شرح مختصری بیان گردید.

پی نوشت:

۱- اسامی واسط ها به زبان لاتین انتخاب شده است تا در هنگام خواندن کتب لاتین و راهنمای نرم افزار مشکلی ایجاد نشود و همچنین نامگذاری با پیشوند I اول کلمه Interface برای خواننده مطلب ملموس گردد.

۲- زبان مدلسازی (UML) Unified Modeling Language فراهم آورنده ابزارهای مدل سازی، مستندسازی، قواعد استاندارد و یکپارچگی نمادگذاری در برنامه نویسی شی گرا است. این زبان مدل سازی قادر است که انواع نمادگذاری مراحل مختلف فرایندهای تولید نرم افزاری را انجام دهد و برخی از کدهای مورد نیاز مدل ها را به زبانهای مختلف برنامه نویسی ایجاد کند. UML یک نمادگذاری است، مجموعه ی از نمودارها و عناصر نموداری است که ممکن است برای شرح طراحی یک سیستم نرم افزاری مرتب شوند. UML فرایند و روشی برای مقایسه یک نمادگذاری و یک فرایند نیست (اندروفیلو، ۱۳۸۴).

۳- در این نرم افزار تغییر و تنظیم میله ابزارها و منوهای کاربرپسند، بدون برنامه نویسی با گزینه سفارشی کردن (Customize) امکان پذیر است. این تغییرات مختصر و ساده، در فارسی سازی منوها و تنظیم فرامین با فعال کردن این گزینه از طریق منوی Tools یا View>Toolbars انجام می گردد.

۴- Event-Driven

۵- IntelliSense

۶- لازم بذکر است که در نرم افزارهای قبلی این شرکت همانند ArcView ۳، زبان برنامه نویسی Avenue بوده است.

۷- Project Explorer

۸- یک برنامه شامل چندین فرم و مدول (ماجول Module) است. هر مدول متشکل از یک یا چندین روال است.

۹- به مجموعه بلوک کد یا یک سری خط-کد برنامه نویسی که کار مشخصی را در نرم افزار انجام می دهد، روال (Procedure) گویند.

منابع:

■ اندروفیلو، ۱۳۸۴، آموزش کاربردی UML (یو، ام، ال) در Studio .NET Visual، مترجم رامین مولاناپور، موسسه فرهنگی هنری دیباگران تهران، تهران.

■ پری، گرگ، ۱۳۷۹، آموزش گام به گام: ویژوال بیسیک ۶ مقدماتی و پیشرفته سادهترین روش یادگیری ویژوال بیسیک در فقط ۲۱ روز، مترجمین فرهاد غنیزاده و سهیلا سرهنگی، مرکز فرهنگی نشرگستر، تهران.

- دیتل، هاروی، پل دیتل، ۱۳۸۴، راهنمای جامع برنامه نویسان ویژوال بیسیک ۶، ترجمه بهرام پاشایی، جهان نو، تهران.
- فرانکین، کیت، ۱۳۸۰، VB.NET برای توسعه گران، ترجمه زهرا بناکار، انتشارات سائسی، تهران.
- کاردان، رحمت اله، جزوه آموزش نرم افزار ArcGIS، موسسه آموزشی پژوهشی وزارت صنایع و معادن، ۱۳۸۱.
- کلارک، ج.، ۱۳۸۴، آموزش گام به گام VBA برای اتوکد ۲۰۰۲، مترجم مرتضی متواضع، موسسه فرهنگی هنری دیباگران تهران، تهران.
- ویژوال بیسیک ۶ Desktop Application، ۱۳۸۱، ترجمه بهرام پاشایی، ادبستان، تهران.
- یوردون، ادوارد و پیترکد، ۱۳۷۸، متدولوژی طراحی شی گرا، ترجمه منصور شریفی کلویی، نشر آروین، تهران.
- ArcGIS® Desktop Developer, 2004, ESRI press, California, USA.
- Burke, R., 2003, GETTING TO KNOW ArcObjects Programming ArcGIS with VBA, ESRI Press, CALIFORNIA.
- Razavi, A. H., 2002, ArcGIS Developer's Guide for VBA, Delmar Learning, Canada.
- Wachowicz, Monica, 1999, Object-Oriented Design for Temporal GIS, Taylor & Francis, Bristol, UK.