

مقالات آموزشی بنیاد سور

آشنایی با زبان برنامه‌نویسی سالی‌دیتی

شبکه بلاگچین سور
www.surnet.org



گردآورنده: بنیاد سور

نسخه: شماره ۳۲





آشنایی با زبان برنامه‌نویسی سالیدیتی

دپ‌ها یا همان برنامه‌های کاربردی غیرمتمرکز (Dapp)، برنامه‌های کامپیوتری هستند که بر روی شبکه‌های بلاکچینی منبع‌باز بی‌واسطه نظیر اتریوم توسعه داده می‌شوند و از قراردادهای هوشمند و واسطه‌های کاربری Front-end استفاده می‌کنند.

توسعه یک دپ، همانند سایر برنامه‌های کاربردی، نیازمند برنامه‌نویسی و اجرای کدها بر روی سیستم هستند. زبان برنامه سالیدیتی، بستری برای توسعه دپ‌ها بر روی شبکه اتریوم است. این زبان برنامه‌نویسی محبوب، توسط تیم اولیه توسعه‌دهنده شبکه بلاکچین اتریوم ایجاد شده است. لازم به ذکر است که زبان‌های برنامه‌نویسی دیگری هم وجود دارند که می‌توان برای ایجاد قراردادهای هوشمند در شبکه اتریوم، از آنها استفاده کرد، اما این روزها انتخاب اکثر برنامه‌نویسان این حوزه سالیدیتی است.

از آنجایی که شبکه سور، در بسیاری از ابعاد فنی مشابه شبکه اتریوم است، در این مقاله به بررسی زبان سالیدیتی پرداختیم، زبانی که به کمک آن می‌توان بر روی شبکه سور، دپ توسعه داد یا دپ‌های توسعه داده شده بر روی شبکه اتریوم را به این شبکه نیز مهاجرت داد.

زبان برنامه‌نویسی سالیدیتی چیست؟

زبان سالیدیتی در سال ۲۰۱۴ توسط گوین وود (Gavin Wood)، یوئچی هیرای (Yuichi Hirai)، کریستین ریتویسنر (Christian Reitwiessner) و الکساندر برگساز (Alex Beregszasz) معرفی شد.

سالیدیتی یک زبان برنامه‌نویسی شی‌گرا است و توسط تیم توسعه‌دهنده شبکه اتریوم برای ایجاد و طراحی قراردادهای هوشمند بر روی این شبکه بنانهاده شده است.



- از این زبان می‌توان برای ایجاد قراردادهای هوشمندی که یک منطق کسب‌وکاری را پیاده‌سازی می‌کند و زنجیره‌ای از تراکنش‌ها را بر روی شبکه بلاکچین تولید می‌کند استفاده کرد.
- این زبان همانند یک ابزار برای ایجاد کد سطح ماشین و کامپایل آن بر روی ماشین مجازی اتریوم عمل می‌کند.
- این زبان شباهت‌های زیادی با زبان برنامه‌نویسی C و C++ دارد و نسبتاً به سادگی می‌توان آن را فرا گرفت. برای نمونه "main" در زبان C معادل "contract" در زبان سالیدیتی است.

همانند سایر زبان‌های برنامه‌نویسی، سالیدیتی نیز دارای انواع متغیر، تابع، کلاس، رشته و دیگر مفاهیم مشابه است. کدهای سالیدیتی شباهت بسیاری به کدهای C++ یا C# و حتی جاوا اسکریپت دارند. برنامه‌نویسان پایتون نیز در هنگام کار با سالیدیتی متوجه می‌شوند که نوع متغیرها در این زبان باید به‌طور صریح تعیین شوند.

تحولات زبان برنامه‌نویسی سالیدیتی

سالیدیتی یک زبان برنامه‌نویسی نسبتاً جدید اما به سرعت در حال رشد است.

- در حال حاضر، سالیدیتی زبان اصلی و محوری شبکه اتریوم و سایر بلاکچین‌های خصوصی مشابه همانند Monex و هایپرلجر بارو است.
- SWIFT یک مدل مفهومی یا PoC بر روی شبکه هایپرلجر بارو و با استفاده از بان سالیدیتی توسعه داده است.

سالیدیتی، به دلایل زیر، زبان محبوب توسعه‌دهندگان برای ایجاد قراردادهای هوشمند است:

- یادگیری و استفاده بسیار آسان
- فراهم کردن شفافیت و دقت بالا



- سرعت، کارایی و امنیت بالای قراردادهای نوشته‌شده با سالیدیتی
- سازگاری کامل با شبکه بلاک چین اتریوم
- زبان برنامه نویسی ایستا (امکان شناسایی خطاها در مرحله توسعه)

اگر بخواهیم به مهم‌ترین کاربرد زبان سالیدیتی اشاره داشته باشیم، باید به این موضوع پردازیم که این زبان تخصصی باعث اتصال بلاک‌های قبلی و بعدی شبکه بلاک چین به یکدیگر می‌شود تا در انتها یک مجموعه زنجیروار و متصل به هم شکل بگیرد؛ با شکل‌گیری چنین شبکه بلاک چینی، هیچ شانسی برای هک شدن یا دستکاری در تراکنش‌ها و معاملات وجود نخواهد داشت.

ویژگی‌های منحصربه‌فرد این زبان که برای بلاک چین اتریوم، امنیت و شفافیت به وجود آورده است، کاربردهای دیگری هم دارد. از جمله کاربردهای دیگر این زبان می‌توان به امکان به کارگیری آن در مزایده‌ها، حراج‌ها و فرآیندهای رای‌گیری اشاره کرد. همچنین، علاوه بر نوشتن و ایجاد کردن قراردادهای هوشمند در اتریوم، امکان توسعه دادن و بهبود شبکه اتریوم را هم فراهم می‌کند. قراردادهای هوشمند یا Smart Contract از دو قسمت مجزا و ویژه دیتا و کد تشکیل شده‌اند و تنها با قرارگیری این زبان تخصصی در آن‌ها به صورت خودکار و بدون دخالت شخص ثالث انجام می‌شوند.

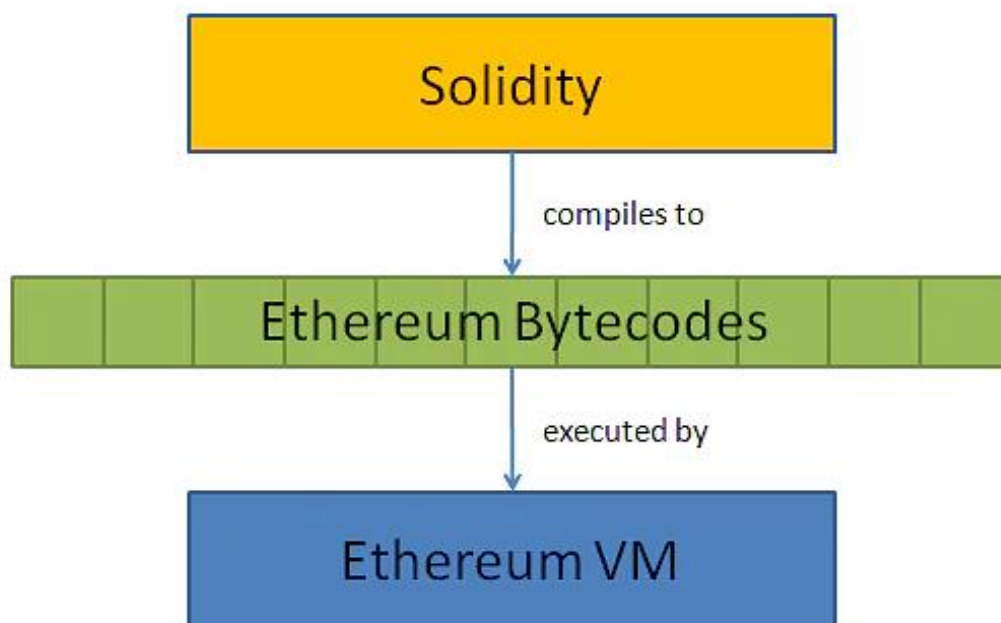
ارتباط سالیدیتی با ماشین مجازی اتریوم و قراردادهای هوشمند

قراردادهای هوشمند مبتنی بر زبان سالیدیتی، روی ماشین مجازی اتریوم (Ethereum Virtual Machine) راه‌اندازی و اجرا می‌شوند. ماشین مجازی اتریوم، یک ماشین تورینگ کامل است. این ماشین یک مدل ریاضیاتی محاسبه است که حافظه و زمان پردازش بی‌نهایت در اختیار دارد.

- ماشین مجازی اتریوم یک محیط اجرایی برای قراردادهای هوشمند اتریومی ایجاد می‌کند.



- وظیفه اصلی آن، تضمین امنیت و اجرای برنامه‌های غیرقابل اعتماد با استفاده از یک شبکه بلاکچین عمومی است.
- ماشین مجازی اتریوم برای جلوگیری از حملات Denial-of-Service تجهیز شده است و تضمین می‌کند برنامه‌ها با اینکه با یکدیگر در ارتباط هستند، به وضعیت یکدیگر دسترسی نخواهند داشت.



یک قرارداد هوشمند در زبان برنامه‌نویسی سالیدیتی توسط مجموعه‌ای از داده‌ها و یک قطعه کد تعریف می‌شود که داخل یک آدرس در شبکه اتریوم قرار گرفته است. تراکنش‌های داخل بلاک در بلاک چین توسط این قرارداد مدیریت می‌شوند.

یک آدرس مشخص در قالب یک کلید رمزنگاری به هر بلاک اختصاص یافته است. این آدرس از طریق اعمال تابع هش روی آدرس بلاک قبلی به وجود می‌آید. به همین دلیل بین بلاک‌های مجاور یک ارتباط ایجاد می‌شود که امکان ایجاد هرگونه تغییر در بلاک‌ها تنها از این روش ممکن است.



قراردادهای هوشمند در واقع برنامه‌های خود-انجام (Self-executing) هستند که انجام معاملات دیجیتالی، تجارت، مبادله ارز، انتقال دارایی و موارد مشابه را با توجه به مجموعه‌ای از شرایط از پیش تعریف‌شده به نام پروتکل قرارداد هوشمند امکان‌پذیر می‌کنند. قراردادهای هوشمند علاوه بر این که باعث اطمینان از پیروی قوانین و شرایط قرارداد توسط دو طرف می‌شوند، تعهدات قرارداد را نیز به‌طور خودکار اجرا می‌کنند.

توسعه‌دهندگان و برنامه‌نویسان در اکثر موارد از زبان برنامه‌نویسی سالیدیتی برای نوشتن قراردادهای هوشمند در بلاک چین اتریوم استفاده می‌کنند. این زبان برنامه‌نویسی به آن‌ها کمک می‌کند تا کدهای سطح بالایی بنویسند که به راحتی در کد ماشین سطح پایین نیز قابل اجرا هستند. در سالیدیتی برای ساده‌سازی فرآیند توسعه، از پیش‌شرط‌ها، پس شرط‌ها و غیر متغیرها استفاده می‌شود. به این ترتیب فرآیند تولید قراردادها برای کاربران بسیار ساده‌تر از سایر زبان‌های برنامه‌نویسی خواهد بود.

- قراردادهای هوشمند کدهای کامپیوتری سطح بالایی هستند که بر روی EVM، پیش از قرارگیری بر روی شبکه اتریوم برای اجرا، کامپایل می‌شوند.
- به کمک قراردادهای هوشمند می‌توان بدون نیاز و درگیر کردن نهادهای سوم، تراکنش‌هایی مطمئن ایجاد کرد. تراکنش‌هایی که قابل رهگیری بوده و قابل بازگشت نیستند.
- زبان‌های برنامه‌نویسی که عموماً برای ایجاد قراردادهای هوشمند استفاده می‌شوند، Solidity، Serpent، Mutan و LLL هستند.



در مقاله بعدی در زمینه دوره‌های آموزشی موجود برای آغاز یادگیری این زبان برنامه نویسی خواهیم پرداخت.