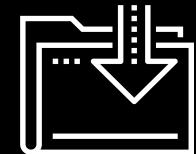




# Token Standards

FinTech  
Lesson 21.2



# Class Objectives

---

By the end of the class, students will be able to:



Differentiate fungible from non-fungible tokens.



Explain that the Ethereum Request for Comments (ERC) standards are official smart contract implementations for various use cases.



Explain which ERC token standards correspond with fungible vs. non-fungible tokens.



Implement the ERC-20 standard to create a fungible token by using the OpenZeppelin library.



Deploy and test a token that uses the ERC-20 standard by using MetaMask and a Ganache blockchain.



Define the terms **initial coin offering (ICO)** and **crowdsale**.

# Fungible and Non-Fungible Tokens

# Fungible and Non-Fungible Tokens

As previously discussed, we can use tokens to represent any potential store of value. These include:

Votes



Currency



Property



Arcade games



# Fungible Tokens

---

Some tokens represent assets that have interchangeable and exchangeable values. These are called **fungible tokens**. Tokens that represent fiat currency or cryptocurrency often fall into this category.



A token that represents one ether or one bitcoin, for example, holds the same value as another token that represents one ether or one bitcoin.

And, these tokens are interchangeable with each other.



**Fiat currency** is a government issued currency that is backed by a central bank, instead of a physical commodity like gold. The majority of modern paper currencies, such as the U.S dollar, are fiat currencies.

# Fungible Tokens

---

As with fiat currencies, an exchange rate exists between cryptocurrencies. Individuals can thus exchange a single bitcoin for the appropriate amount of ether.

Currency	Average rate	Exchange (24h)
Bitcoin (BTC)	49 657.00 USD	3.69%
Ethereum (ETH)	3 405.63 USD	1.27%
Cardano (ADA)	2.23 USD	1.82%
Tether (USDT)	1.00 USD	0.05%

**Note:** This data may change over time.

# Fungible Tokens

---

Tokens that represent other types of assets can also be fungible.



## For example:

Consider tokens that represent votes.

Any individual's vote holds power that is equal to that of any other individual's vote.

So, the tokens that represent their votes have equal values.

The practice of representing assets as tokens on a decentralized blockchain is called tokenomics.

This is a powerful concept.

But, it goes deeper than representing the values of interchangeable assets.



# Non-Fungible Tokens

---

Tokens that represent stores of value that aren't directly comparable and interchangeable are called **non-fungible tokens**, or **NFTs**.



- NFTs depend more on which assets someone owns rather than on how many.
- The values of NFTs are more difficult to determine.
- They're not interchangeable with other NFTs.

# Non-Fungible Goods

Examples include:

Parcels of land



Diamonds



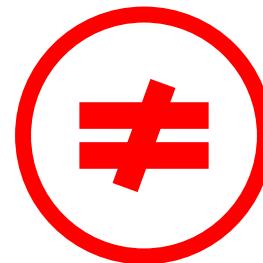
Collectibles



# Non-Fungible Goods

---

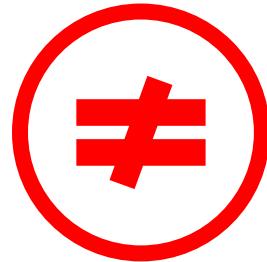
A person might own one parcel of land in California and another in Pennsylvania. These parcels differ, so they don't hold the same value. So, they are not interchangeable.



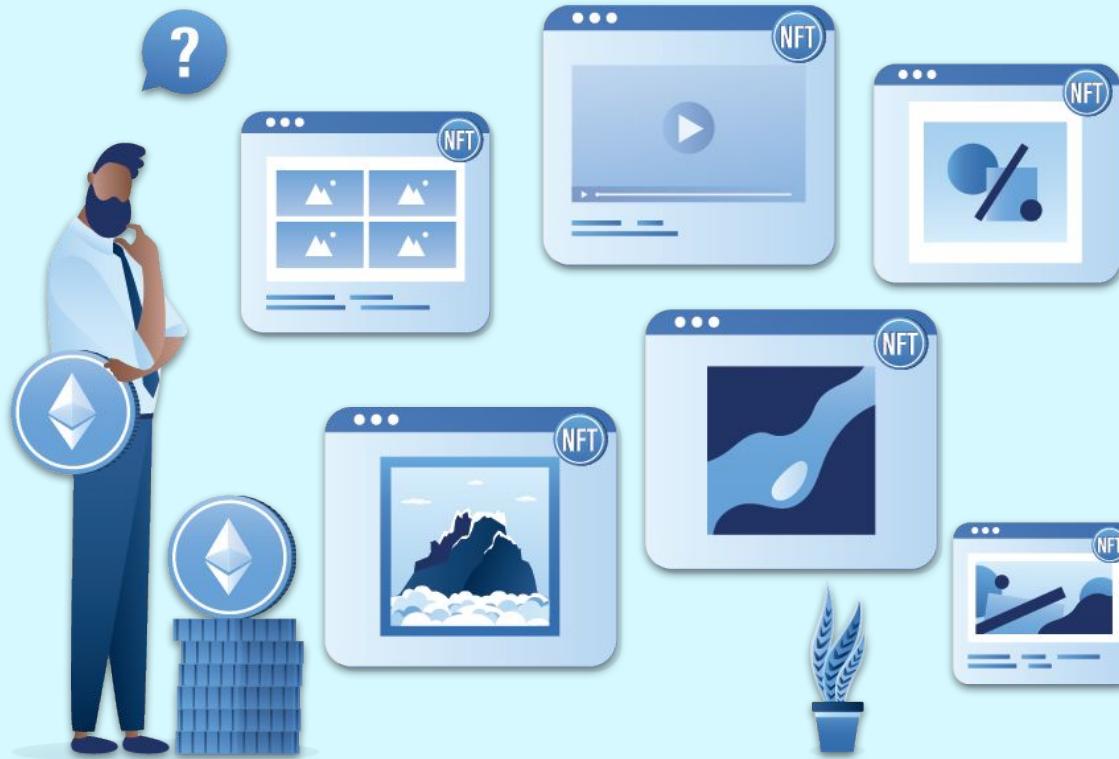
# Non-Fungible Goods

---

The same is true for two collectible figures, like a Pop Vinyl and a Beanie Baby. Each collectible is unique.

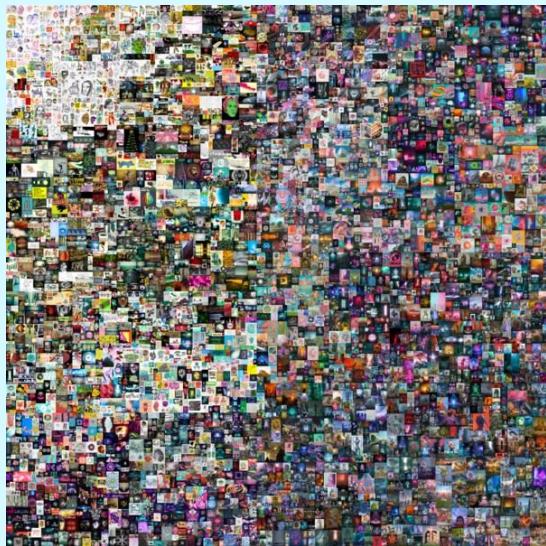


Let's discuss some of the real-world implementations of NFTs on the blockchain.



# Non-Fungible Tokens

NFTs have recently gained a lot of popularity among artists and art collectors.



Christie's recent sale of an NFT by the artist Beeple for \$69 million set a new record for digital art.

just setting up my twtr

4:50 AM · Mar 22, 2006

151.6K 126.8K Copy link to Tweet

Want to buy this tweet?

The highest offer is \$2500000 by [@sinaEstavi](#)

\$ 0.00 (0.0000)

OFFER

Counter-offer must be a minimum increase of \$1 or 10%, whichever is more.

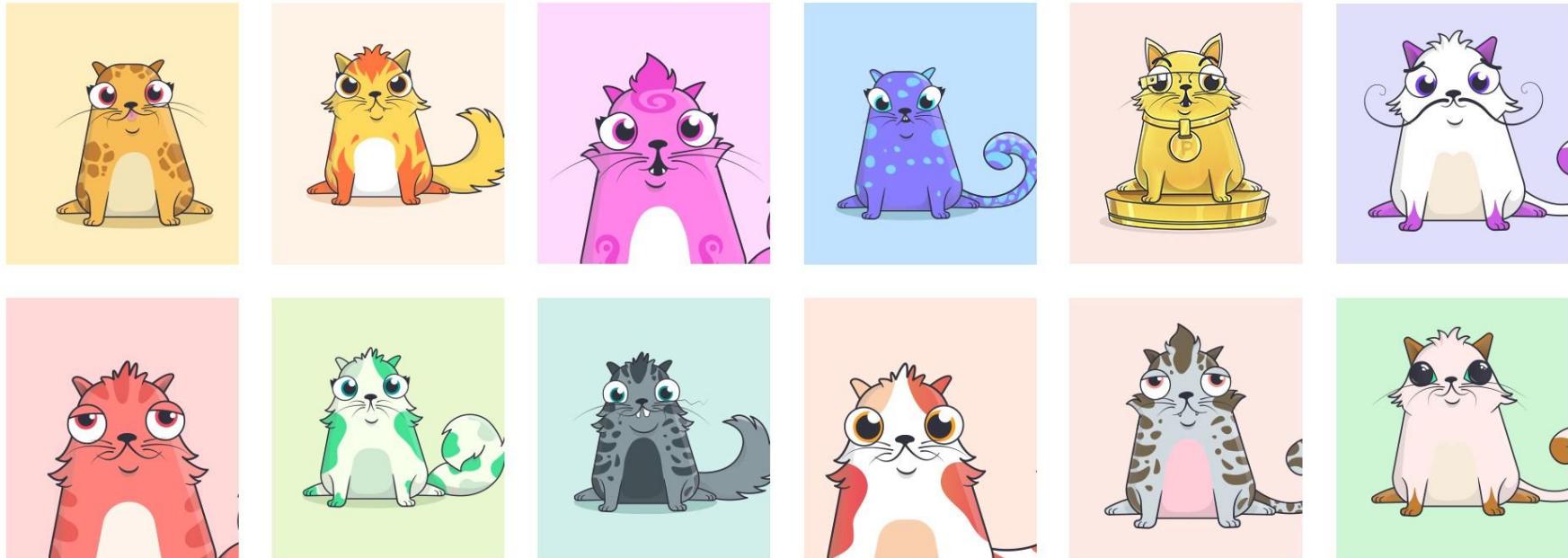
Jack Dorsey, the founder of Twitter, auctioned off an NFT that represents his first tweet for \$2.9 million.

CryptoKitties is a popular NFT application on the Ethereum blockchain.



# CryptoKitties

Users can buy, sell, and breed virtual cats in the form of NFTs.





**At the cryptocurrency peak in 2017, some cryptokitties were valued at over \$130,000 USD.**

# Fungible vs. Non-Fungible Tokens

---

NFTs, such as cryptokitties, fundamentally differ from fungible tokens.



Both types have value, but the value of each NFT is unique. This leads to different sets of rules, processes, and implementations for fungible and non-fungible tokens.



The smart contracts that create NFTs follow different standards than those that create fungible tokens.



Today, we'll discuss and use some of the standards that the Ethereum community has created for implementing both fungible and non-fungible tokens.

# Token Standards and ERC-20

# Token Standards

# Token Standards

---

Ethereum continues to grow and rapidly evolve as the blockchain community expands on the concepts of Web 3.0 and blockchain development.

During periods of change, developers form standards that outline best practices for the platform to prevent bugs and security vulnerabilities.

In Ethereum, these standards are known as Ethereum Improvement Proposals, or EIPs.



# Ethereum Foundation

---

The Ethereum Foundation officially refers to EIPs as:

“A design document providing information to the Ethereum community, or describing a new feature for Ethereum or its processes or environment.”



# Token Standards

---

In an earlier module, we used Bitcoin Improvement Proposals (BIPs) to generate a mnemonic seed phrase and to then transform our seed phrase into a private key for our blockchain account.



EIPs resemble BIPs. But, the proposed standard in an EIP is developed specifically for the Ethereum blockchain rather than for the Bitcoin blockchain.



Although there are several categories of EIPs, we'll focus on Ethereum Request for Comments (ERC).

## **Ethereum Request for Comment**

**(ERC)** is a type of EIP that sets standards at the application level.

This means that the standards apply to the code for applications, such as smart contracts and tokens.

# Token Standards

Most EIPs, including ERCs, follow a similar workflow, which includes the following stages:

<b>Work in progress (WIP)</b>	An EIP creator formulates an EIP and can ask for input on community forums.
<b>Draft</b>	The initial EIP draft and any changes get merged into the Ethereum EIP GitHub repository via a pull request. The EIP must then be implemented or, written as code, progress to the next phase.
<b>Last Call</b>	The EIP gets listed on the <a href="#">Ethereum Improvement Proposals</a> website in the Last Call section. If no unaddressed technical complaints or required changes to the source material exist, the EIP will become final. This means that the standard won't change after this point.
<b>Final</b>	The Ethereum blockchain and its participants begin using the processes that the EIP lays out.



Class Slack Channel:  
Some ERCS in various stages of development



Today's class will focus  
on implementing the  
ERC-20 token standard.

We'll focus on the ERC-721 standard in a later unit.



ERC-20

# ERC-20

---

Although other standards apply to fungible tokens, we'll use ERC-20 for the following reasons:



Compared to some other standards, it's simpler to understand and implement.



It has the widest adoption and support among blockchain explorers, wallets, and developer libraries.



ERC-20 tokens are common in the real world. Many popular stablecoins implement them, including USD Coin (USDC) by Coinbase and Gemini dollar (GUSD) by Gemini.



Unlike some other standards for fungible tokens, ERC-20 doesn't require us to incorporate any additional EIPs.

# ERC-20

---

ERC-20 essentially defines the code for a simple smart contract for a fungible token.



Using standards to write contracts ensures consistency and predictability across various contracts.

One ERC-20 token contract should work just like any other ERC-20 token contract.

This is particularly important in the finance realm.

Using standards also makes it easier to write, use, and build applications.

This is because we know which functions are available for use in the applications and how they'll behave.



# ERC-20

---

We use the ERC-20 standard for fungible Ethereum tokens that store value.  
The ERC-20 standard:

01

Defines rules for:

- how the tokens get used
- how transactions get made
- how new tokens get created  
(or minted)

02

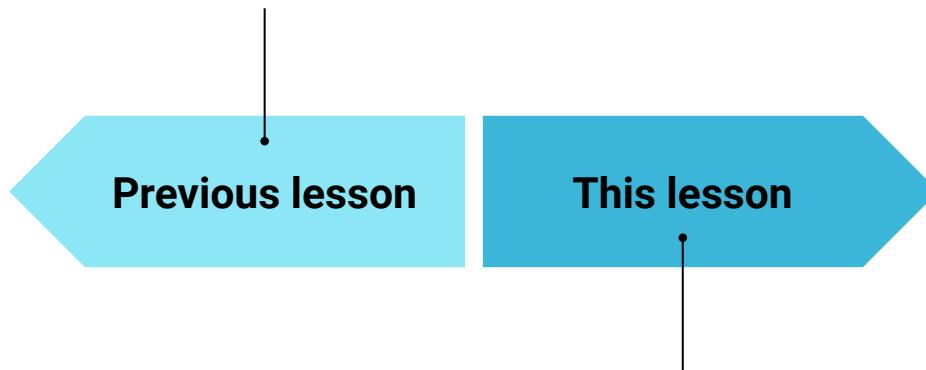
Serves as the basis for many cryptocurrencies and coins, especially stablecoins, that exist today including:

- Chainlink (LINK)
- Wrapped Bitcoin (WBTC)
- Binance Coin (BNB)
- USD Coin (USDC)
- Dai (DAI)

# ERC-20

---

In the previous lesson, we created simple tokens using a basic smart contract.



## NOTE

This will not only make our tokens more secure but also save us some work!

In this lesson, we'll instead learn how to use community-accepted ERC-20 standards to implement fungible tokens.

# ERC-20

---

By keeping track of the most recently accepted EIPs, we can stay up to date on the current Ethereum standards, practices, and technologies.

## Living

Number	Title	Author
1	EIP Purpose and Guidelines	Martin Becze, Hudson Jameson, et al.

## Final

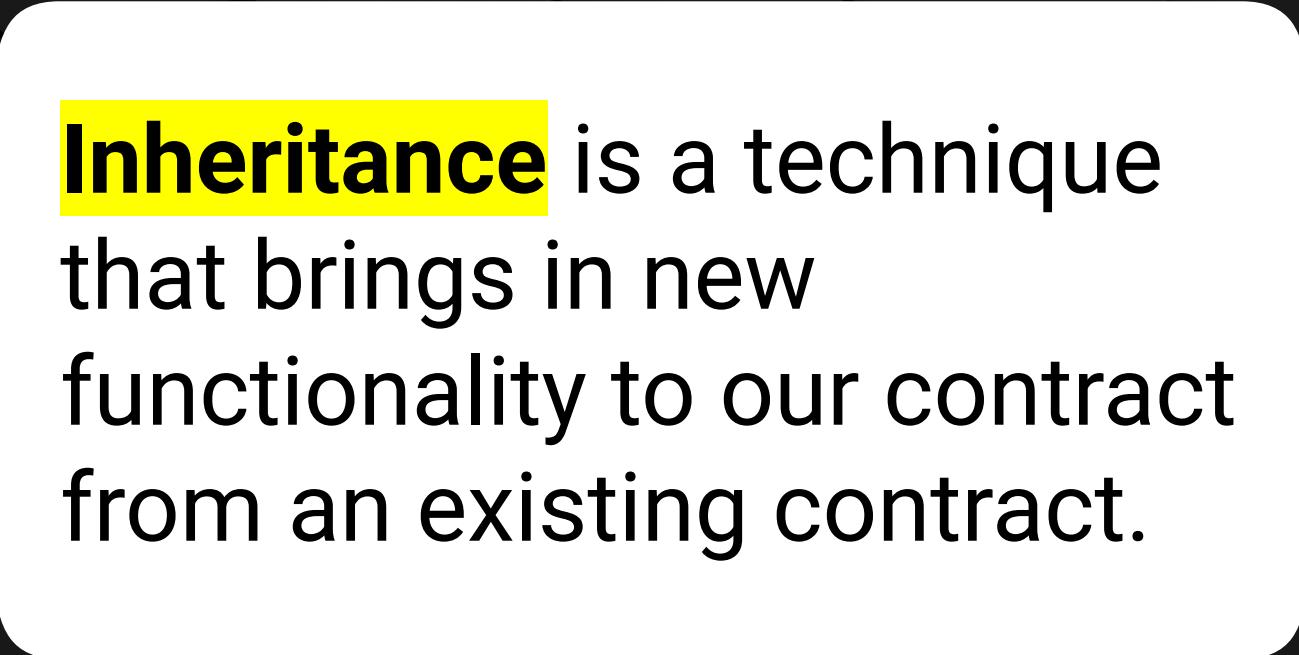
Number	Title	Author
2	Homestead Hard-fork Changes	Vitalik Buterin



## Instructor Demonstration

---

# OpenZeppelin ERC-20 Contracts



**Inheritance** is a technique  
that brings in new  
functionality to our contract  
from an existing contract.

# Questions?





# Activity: ERC-20 Arcade Token

In this activity, you'll create an `ArcadeToken` smart contract by using the `ERC20` and `ERC20Detailed` smart contracts that the OpenZeppelin library provides.

Suggested Time:

---

30 minutes



Time's Up! Let's Review.

# Questions?

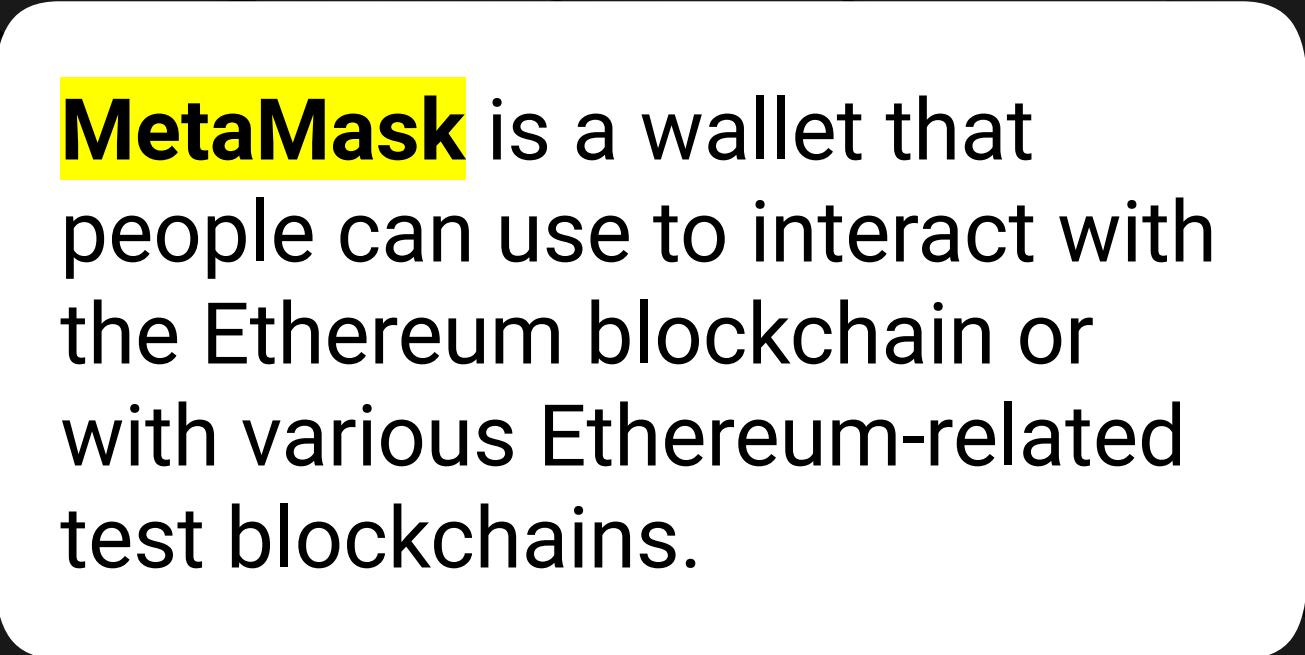


*Break*





# Introducing MetaMask



**MetaMask** is a wallet that people can use to interact with the Ethereum blockchain or with various Ethereum-related test blockchains.

# Tools for Deploying and Testing Contracts

---

The purpose of this demonstration is to understand the role of each application, how each application operates, and how they work together in the contract deployment and testing process.



remix



Ganache



METAMASK

# Introducing MetaMask

---

It's possible to deploy the `ArcadeToken` contract on the main Ethereum blockchain network, which is also called the `mainnet`.



Because the `ArcadeToken` contract uses industry standards, we can deploy it to the main Ethereum blockchain network (the `mainnet`).



In this case, we can have our “ARCD” token listed on a cryptocurrency exchange, or a public marketplace where people can trade tokens



Many wallets and block explorers could then automatically detect it.

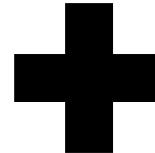


Like all applications (regardless of whether we wrote them in Solidity or Python), it's always a good idea to deploy and test a smart contract to catch any potential issues.

# Introducing MetaMask

---

As we found in an earlier lesson, we can deploy a contract to the Kovan testnet. But it's even easier to deploy and test a contract locally. To do that, we can use a test blockchain application called Ganache and an integrated wallet called MetaMask.



Ganache

METAMASK

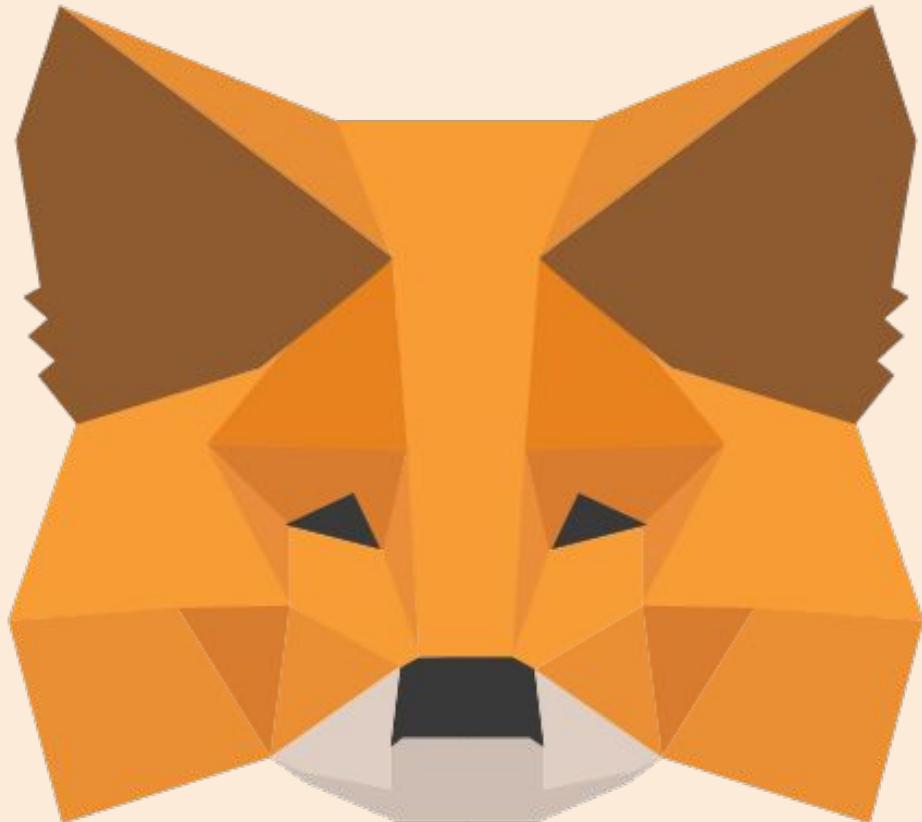
# Introducing MetaMask

---

MetaMask functions as the go-between, routing information between Ganache and Remix.

MetaMask stores the information that's associated with each account.

This includes the private key, the address, the ether balance, and the balance of other tokens that the account has purchased (including the **ArcadeToken** if set up properly).



# Introducing Ganache

Ganache is an application that launches a personal blockchain for the development and testing of applications that are designed for use on the Ethereum blockchain.

With Ganache, a user can develop, deploy, and test their decentralized applications in a safe environment.

In effect, Ganache is a personal testnet.



# Recap

---



remix

We use Remix to develop, compile, deploy, and test the contract operations.



Ganache

Ganache mimics the Ethereum blockchain but in the local environment. Ganache creates the account that we use to test the application. It also creates a record of the transactions that occur in the contract.



METAMASK

MetaMask is the wallet. It functions as the go-between, routing information between Ganache and Remix. MetaMask stores the information that's associated with each account.

# Questions?





# Deploying Tokens with Ganache and MetaMask

Suggested Time:

---

20 minutes



# Activity: Build and Deploy XP\_Token

In this activity, you'll first implement `XP_Token` by using the `ERC20` contract that the OpenZeppelin library provides. They'll then deploy and test the `XP_Token` functionality by using Ganache and MetaMask.

Suggested Time:

---

30 minutes



Time's Up! Let's Review.

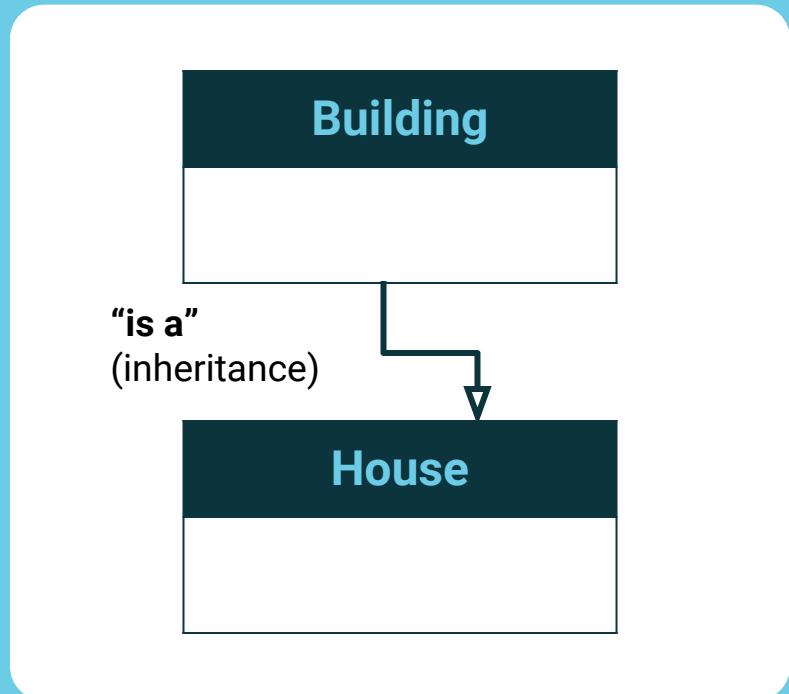


Which OOP concept influences the way that the ERC20 and ERC20Detailed contracts get applied to the XP\_Token contract?

# Answer

This OOP concept is inheritance.

- The concept of inheritance proves useful when a certain type of relationship exists between two classes.
- Specifically, we can use inheritance when we consider one class to be a specialized version of the other, more-general one.
- In this case, **ERC20** is the general class. And **XP\_Token** is the specialized version of that class.





What's a constructor, and what  
role does it play in the  
**XP\_Token** contract?

A **constructor** is a specialized function that we can use to pass initial values to our contract.

---

In the case of XP\_Token, we can use the constructor to pass values for the `initial_supply` of tokens, the `name`, the `symbol`, and the number of `decimals` that are associated with the token distribution.



What's significant about the value of 18 that leads to its selection as the **decimals** number in our constructor?

# Answer

---

The smallest denomination of either is the wei.

Each wei is an 18-decimal-place fraction of an ether.

wei	Ether value
1	$10^{-18}$ ETH

Also, setting the decimals value of XP\_Token to 18 means that each wei can buy one XP\_Token.

This greatly simplifies the token calculation and distribution.



# What's Ganache?

# Answer

Ganache is a desktop application that developers can use to locally implement a test version of the Ethereum blockchain.

ACCOUNTS	BLOCKS	TRANSACTIONS	CONTRACTS	EVENTS	LOGS	SEARCH FOR BLOCK NUMBERS OR TX HASHES
CURRENT BLOCK 0	GAS PRICE 20000000000	GAS LIMIT 6721975	HARDFORK MUIRGLACIER	NETWORK ID 5777	RPC SERVER HTTP://127.0.0.1:7545	MINING STATUS AUTOMINING
MNEMONIC <small>?</small> ship oval acoustic symptom lock lounge pass long solution body remain conduct						HD PATH m/44'/60'/0'/0/account_index
ADDRESS <a href="#">0x12ACC79F94D98d7F212ac4cfA942fF6fDFE6B215</a>	BALANCE 100.00 ETH			TX COUNT 0	INDEX 0	<a href="#">🔗</a>
ADDRESS <a href="#">0x2FD589203132415ef1c9776A2eb6eBa2117f8b56</a>	BALANCE 100.00 ETH			TX COUNT 0	INDEX 1	<a href="#">🔗</a>
ADDRESS <a href="#">0xf878543ba3246d7cF0D24afEa7c6753e8300F3AA</a>	BALANCE 100.00 ETH			TX COUNT 0	INDEX 2	<a href="#">🔗</a>
ADDRESS <a href="#">0xCEe844E50bcB2BdEBA7D20F4b84be88c8D30E156</a>	BALANCE 100.00 ETH			TX COUNT 0	INDEX 3	<a href="#">🔗</a>
ADDRESS <a href="#">0x90e6b33fd2A7350f40DA55E7109c8B60F4065d6f</a>	BALANCE 100.00 ETH			TX COUNT 0	INDEX 4	<a href="#">🔗</a>
ADDRESS <a href="#">0x596204aD82Eee4C33401a05eCe182975561eBd92</a>	BALANCE 100.00 ETH			TX COUNT 0	INDEX 5	<a href="#">🔗</a>
ADDRESS <a href="#">0x4B990FF4567f252ddb37d67278260dA9e2a9260C</a>	BALANCE 100.00 ETH			TX COUNT 0	INDEX 6	<a href="#">🔗</a>
ADDRESS <a href="#">0x6a5bE5585ba31727D381D59635579E219d7B2472</a>	BALANCE 100.00 ETH			TX COUNT 0	INDEX 7	<a href="#">🔗</a>
ADDRESS <a href="#">0xf088Ff4F826AA631a27ADFE752931Fc397E4cBeB</a>	BALANCE 100.00 ETH			TX COUNT 0	INDEX 8	<a href="#">🔗</a>
ADDRESS <a href="#">0xE3b18385D399fD42Cb14751bc92AB3926a9425e0</a>	BALANCE 100.00 ETH			TX COUNT 0	INDEX 9	<a href="#">🔗</a>



How many times do you need  
to set up the Ganache network  
in MetaMask?

# Answer

---

You need to set up the Ganache network only once in MetaMask.

That's because the RPC server location for Ganache doesn't change.

(HTTP://127.0.0.1:7545)



# Questions?



# Real-World Token Examples

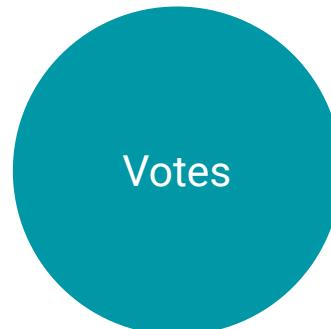


# What things can a token represent?

# Real-World Token Examples

---

Things that can be represented by a **token**:





What are some common aspects  
of a token?

# Real-World Token Examples

---

Some common aspects of a **token**:





What are some generally accepted differences between coins and tokens?

# Real-World Token Examples

Some generally accepted differences between **coins** and **tokens**:

Coins	Tokens
Commonly make up a fundamental component of a blockchain.	Commonly built on an existing blockchain.
Typically used just as a currency for buying and selling things.	Can have broader use cases.



Sometimes, they don't differ. Many people in crypto communities use the terms interchangeably.



What are some potential benefits  
of tokenizing an asset on an  
open blockchain?

# Real-World Token Examples

---

Some potential benefits of tokenizing an asset on an open blockchain:

01

The ability to globally trade the asset without any additional infrastructure.

02

The easy transfer of ownership with improvements in liquidity and auditability.

03

The open blockchain benefits: open, public, borderless, censorship resistant, and neutral.



What is the difference between  
fungible and non-fungible tokens?

# Real-World Token Examples

---

## Fungible tokens

- With fungible tokens, we ask “how many.”
- Fungible tokens are interchangeable. Each token can be exchanged with another token of the same value.

## Non-fungible tokens

- With NFTs, we ask “which one.”
- Non-fungible tokens are unique, per-token assets. Each token has its own properties and value, like artwork and legal certificates. An NFT can't be exchanged with another NFT.

# Discussion

# Discussion

---

Blockchain and the use of coins and tokens are changing the way that that world conducts transactions.



# Discussion

Blockchain has the ability to revolutionize any industry that involves either transactions or record-keeping.

## These include:

- Finance and investing
- Medicine
- Shipping and receiving
- Logistics



## Discussion

---

One of the most common uses cases for fungible tokens—also known as coins or stablecoins—is finance. The term for these evolving financial systems that use fungible tokens is **decentralized finance**, or **DeFi**.

DeFi

*is short for*

Decentralized Finance

# Discussion: DeFi

Firms building DeFi applications use smart contract functionality to perform many traditional financial services. Rather than conducting their business by using fiat currency (such as the US dollar, the Euro, or the Chinese Yuan), DeFi applications all use blockchain tokens, coins or, most likely, stablecoins.

Examples of coins



Dai  
by MakerDao



COMP  
by Compound

Examples of stablecoins



Tether  
(USDT)



USD Coin  
(USDC)



Binance USD  
(BUSD)

# Discussion: DeFi

Companies operating in the DeFi space include:

Payments  
processors



Lending  
firms



Decentralized  
assets exchanges



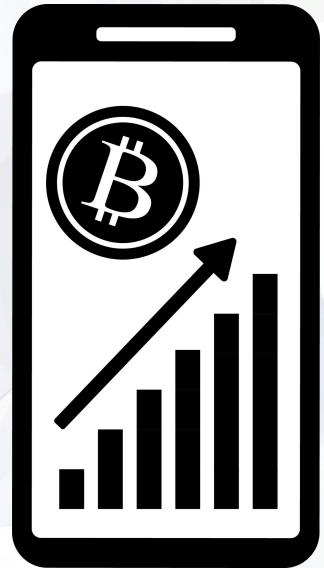
Decentralized  
derivative exchanges



Blockchain and smart contracts underlie DeFi applications, which are anchored on the concept of

---

**Composability:** fintech entrepreneurs can use the public functions of existing smart contracts to create, or compose, new smart contracts.



Blockchain tokens are gaining popularity in areas well beyond financial services.

# Discussion: BAT

---

Brave is a security- and privacy-focused internet browser that issues Basic Attention Tokens (BATs).



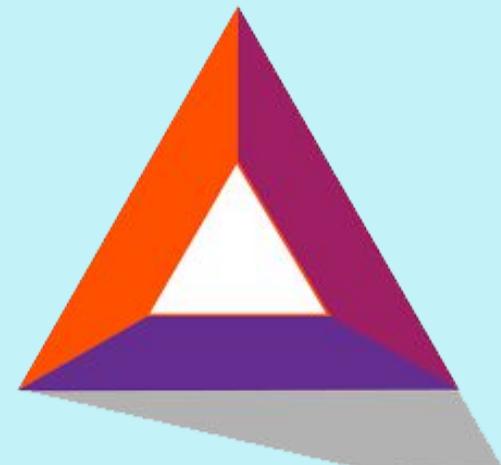
The goal is to revolutionize the digital advertising industry.



A user earns BATs for using the Brave browsing software to surf the web or to display specific advertisements.



Users can spend the BATs to buy products or to tip content creators with Brave Rewards.



# Discussion: GLM

---

Golem is a company seeking to create an economic system that allows individuals to rent excess computing power.



Participants in the network can earn [Golem tokens \(GLMs\)](#) for renting idle digital resources.



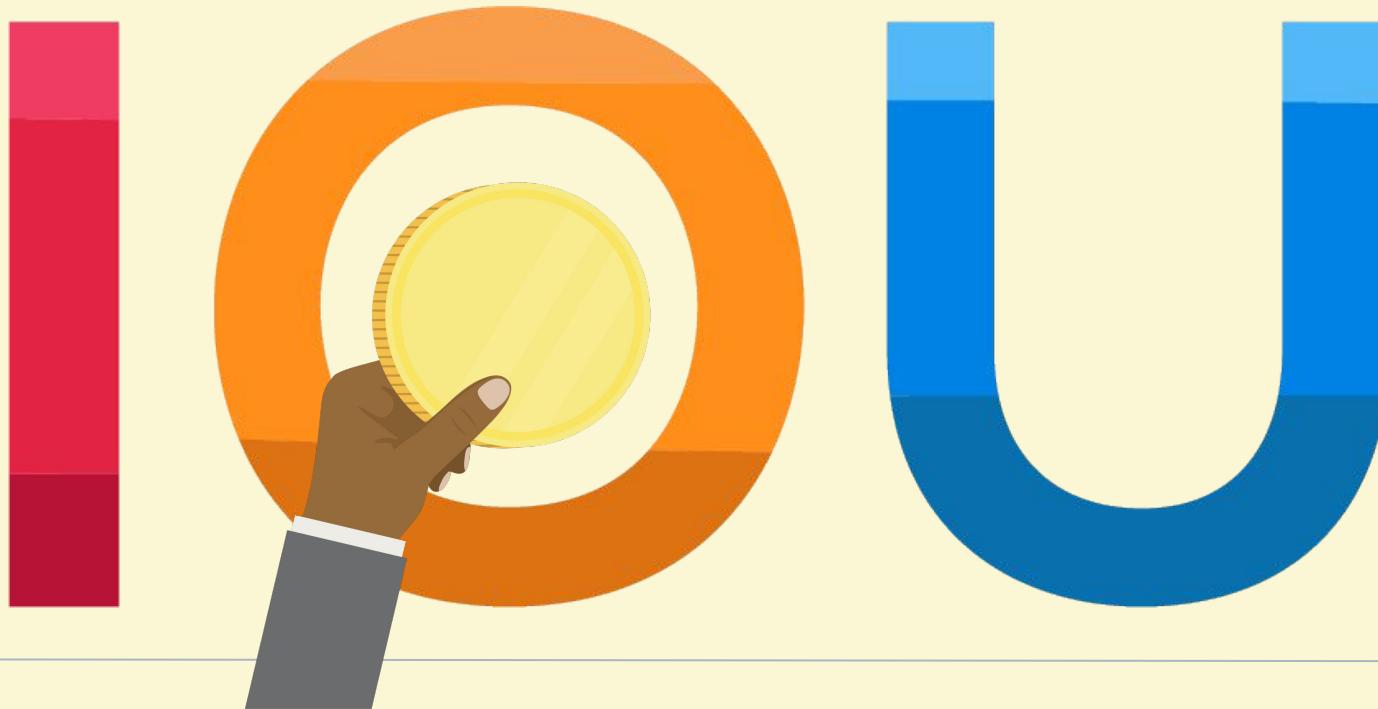
Participants can then use GLM tokens to purchase computing resources when needed.



# Discussion: BAT and GLM

---

We refer to tokens like the BAT and the GLM as utility tokens. People can redeem utility tokens in the future to access the product or service of the issuing company. We do not consider utility tokens as financial investments but more as blockchain-based IOUs.



# Discussion: ICO

Utility tokens and other fungible tokens and coins get issued through an **initial coin offering**, or **ICO**.

An ICO is a fund-raising mechanism, where new coins or tokens are offered in exchange for more-popular cryptocurrencies, like Bitcoin or ether.



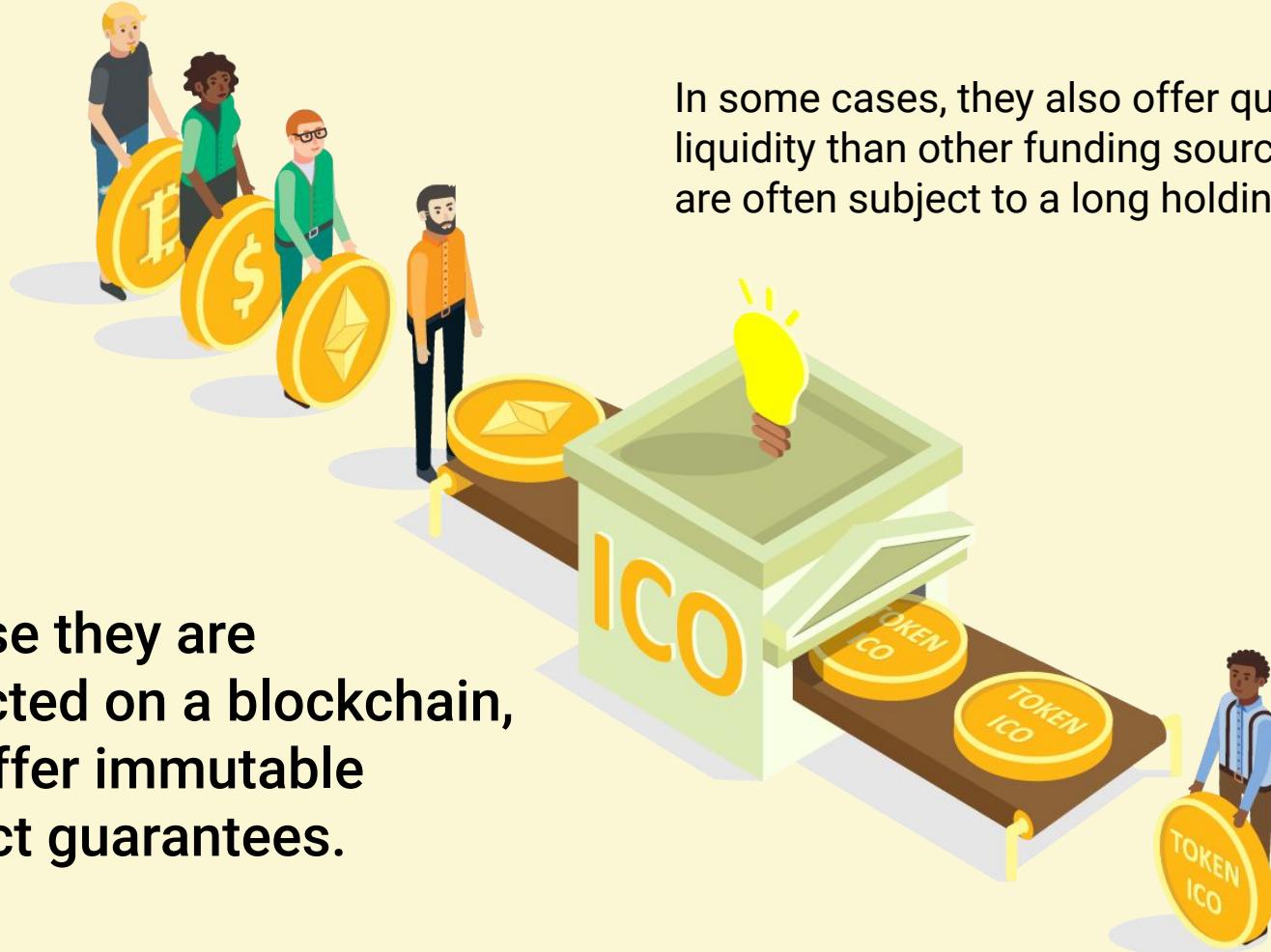
# Discussion: Crowdsale

A **crowdsale** is a blockchain-focused process for capitalizing or financing projects for startups or other companies. Rather than issuing shares in a company, like an initial public offering (IPO) does, an ICO issues tokens.

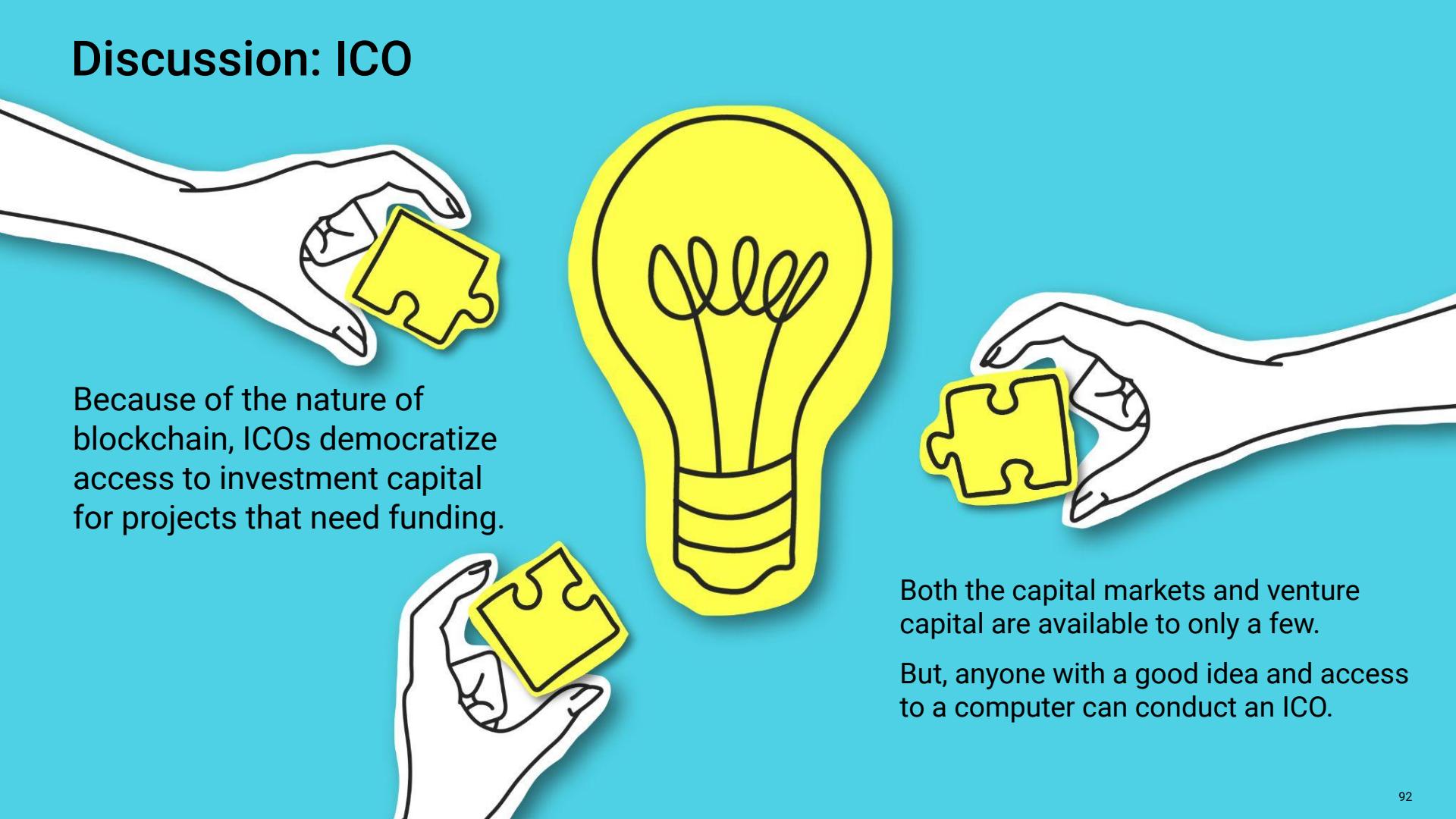


**Because they are conducted on a blockchain, ICOs offer immutable contract guarantees.**

In some cases, they also offer quicker liquidity than other funding sources, which are often subject to a long holding period.



# Discussion: ICO



Because of the nature of blockchain, ICOs democratize access to investment capital for projects that need funding.

Both the capital markets and venture capital are available to only a few.

But, anyone with a good idea and access to a computer can conduct an ICO.

# Next Class

---

Our next class will focus on updating our `ArcadeToken` so that it can get distributed via a crowdsale. We'll use the ERC-721 standard to accomplish that.



# Questions?



The  
End