STUDY MATERIAL OF CLASS 7

1. INTRODUCTION TO DATA: IMAGES, NUMBERS, VOICE

In the digital age, **data** is the most essential building block of information systems, artificial intelligence, machine learning, and user-centric technologies. Understanding the various **types of data** is key to designing smart applications and AI-driven solutions.

1.1. Image Data

- **Definition**: Image data consists of visual information represented in the form of pixels. Each pixel has color and intensity information.
- **Formats**: JPEG, PNG, BMP, SVG, WebP
- **Use Cases**:
  - Face detection in security systems (e.g., CCTV with AI)
  - Medical image analysis (e.g., CT scans, X-rays)
  - Image recognition in apps (e.g., Google Lens)
  - Social media filters and AR effects
- **AI Application**:
  - Convolutional Neural Networks (CNNs) are widely used in AI to process and understand image data.
  - Image classification, object detection, segmentation.

1.2. Numerical Data

- **Definition**: Data that is measurable and represented in the form of numbers.
- **Types**:
  - **Discrete**: Countable values (e.g., number of clicks, transactions)
  - **Continuous**: Measurable values (e.g., temperature, speed)
- **Formats**: CSV, Excel, JSON (with numeric entries)
- **Use Cases**:
  - Financial reports, stock market analysis
  - IoT sensor data (e.g., humidity, motion, voltage)
  - AI-based predictive modeling

- **AI Application**:
  - Numerical data is the backbone of supervised learning algorithms such as Linear Regression, Decision Trees, and Neural Networks.
  - Used for analytics, fraud detection, recommendation engines, and more.

1.3. Voice Data (Audio Data)

- **Definition**: Data derived from sound waves, especially human speech.
- **Formats**: MP3, WAV, FLAC, OGG
- **Use Cases**:
  - Virtual assistants (e.g., Alexa, Google Assistant)
  - Call center automation using speech-to-text
  - Voice biometric authentication
  - Podcast or meeting transcription
- **AI Application**:
  - Natural Language Processing (NLP) and Speech Recognition models (like Whisper by OpenAI)
  - Audio classification (e.g., music genre detection, emotion detection)
  - Text-to-Speech (TTS) and Voice Cloning using Deep Learning

1.4. Unified View of Data Types

Data Type	Nature	AI Techniques Used	Key Applications
Image	Visual	CNN, GAN	Surveillance, Healthcare, AR/VR
Number	Structured	Regression, SVM	Finance, IoT, Statistics
Voice	Temporal	RNN, NLP, TTS, ASR	Virtual Assistants, Accessibility Tools

1.5. Why This Matters

Understanding the type of data:

- Helps in choosing the right **AI model and tools**.
- Aids in **data preprocessing** (e.g., normalization, feature extraction).
- Determines **storage and transmission** methods (images need compression, audio needs sampling, numbers may need encryption).
- Allows for **multimodal AI systems** that use a combination of voice, image, and text (e.g., smart assistants that respond to speech and display images).

2. AI IN DAILY LIFE

Artificial Intelligence (AI) is no longer confined to research labs or sci-fi stories. It is seamlessly integrated into our everyday lives—enhancing comfort, convenience, and capabilities. AI systems now assist us in **creating multimedia, automating tasks, playing games, shopping, communicating**, and more.

2.1 Working with AI Tools for Multimedia Content

AI revolutionizes how we **create, edit, and enhance multimedia content** (text, images, video, audio). It automates creativity and speeds up content development.

A. AI in Image Creation and Editing

- **Tools**:
  - **DALL·E** (by OpenAI): Generates images from text prompts (Text-to-Image).
  - **Canva AI**, **Adobe Firefly**: Design assistance, background removal, auto enhancement.
- **Use Cases**:
  - Designing social media posts
  - Creating logos, banners, and product mockups
  - Generating realistic faces or artistic styles

B. AI in Video Generation

- **Tools**:
  - **RunwayML**, **Synthesia**: Create videos from text, avatars speaking various languages
  - **Pictory**, **Lumen5**: Auto-generate videos from blogs/articles
- **Use Cases**:
  - Marketing videos
  - Explainers, educational videos
  - Virtual influencers

C. AI in Audio and Voice

- **Tools**:
  - **Descript**, **ElevenLabs**, **Murf.ai**: Voice cloning, text-to-speech (TTS), podcast editing
- **Use Cases**:
  - Voiceovers for videos
  - Dubbing in multiple languages
  - Accessibility tools (screen readers)

D. AI in Text Content Creation

- **Tools**:
  - **ChatGPT**, **Copy.ai**, **Jasper AI**
- **Use Cases**:
  - Blogging, SEO content
  - Email writing, ad copy
  - Script generation for YouTube and podcasts

2.2 Game Building Using AI

AI plays a vital role in both **developing and enhancing games**, by automating design, improving intelligence, and customizing gameplay.

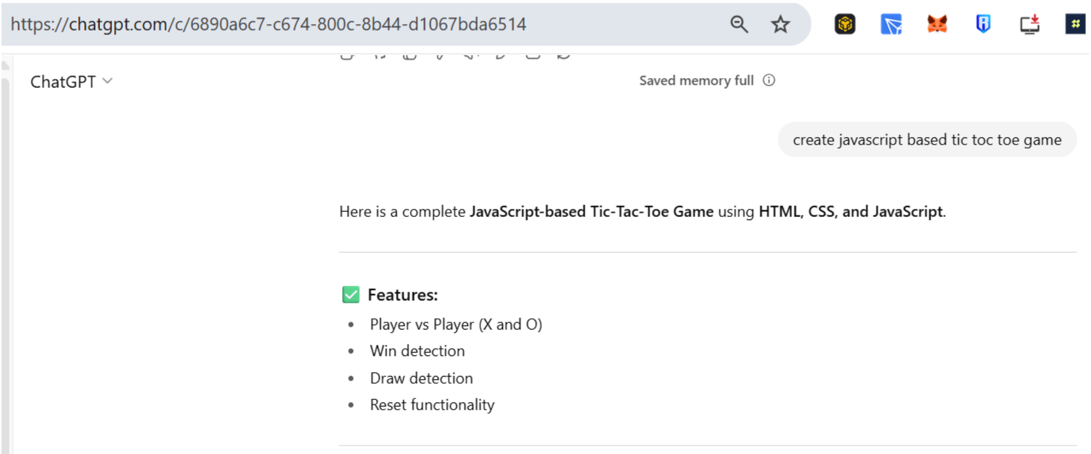
A. AI in Game Development

Students could make use of Chatgpt to create Javascript based games.

B. Tools and Frameworks for AI in Games

1. **ChatGpt**
2. **Webbrowser**
3. **Text Editor (notepad)**

C. Sample



create javascript based tic toc toe game

Here is a complete **JavaScript-based Tic-Tac-Toe Game** using **HTML, CSS, and JavaScript**.

✅ **Features:**

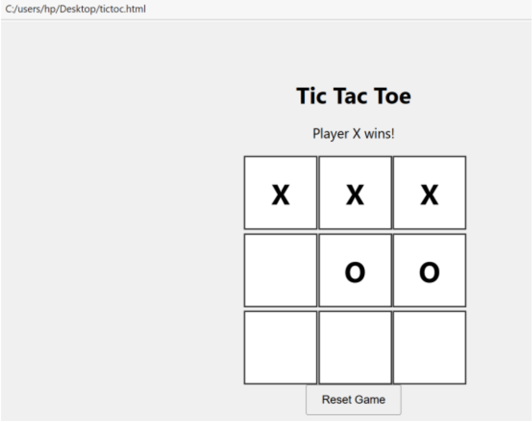
- Player vs Player (X and O)
- Win detection
- Draw detection
- Reset functionality

Fig 1 Prompting for game development

AI Genrated code….

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
  <title>Tic Tac Toe</title>
  <style>
```

```
    body {
      font-family: 'Segoe UI', sans-serif;
      display: flex;
      justify-content: center;
      align-items: center;
      height: 100vh;
      background-color: #f0f0f0;
    }

    .game-container {
      text-align: center;
    }

    h1 {
      margin-bottom: 20px;
    }

    .board {
      display: grid;
      grid-template-columns: repeat(3, 100px);
      grid-gap: 5px;
    }

    .cell {
      width: 100px;
      height: 100px;
      background-color: white;
      border: 2px solid #333;
      font-size: 2.5rem;
      font-weight: bold;
      display: flex;
      justify-content: center;
      align-items: center;
      cursor: pointer;
    }

    .status {
      margin: 20px 0;
      font-size: 1.2rem;
    }

    button {
      padding: 10px 20px;
      font-size: 1rem;
      cursor: pointer;
    }
  </style>
</head>
<body>
  <div class="game-container">
    <h1>Tic Tac Toe</h1>
    <div class="status" id="status">Player X's turn</div>
    <div class="board" id="board"></div>
    <button onclick="resetGame()">Reset Game</button>
  </div>

  <script>
    const board = document.getElementById("board");
    const statusDisplay = document.getElementById("status");
    let currentPlayer = "X";
    let gameActive = true;
    let cells = Array(9).fill("");

    const winningCombinations = [
      [0, 1, 2], [3, 4, 5], [6, 7, 8],  // Rows
      [0, 3, 6], [1, 4, 7], [2, 5, 8],  // Columns
      [0, 4, 8], [2, 4, 6]              // Diagonals
    ];

    function createBoard() {
      board.innerHTML = '';
      cells.forEach((_, index) => {
        const cell = document.createElement("div");
        cell.classList.add("cell");
        cell.dataset.index = index;
        cell.addEventListener("click", handleClick);
        board.appendChild(cell);
      });
    }

    function handleClick(e) {
      const index = e.target.dataset.index;
      if (!gameActive || cells[index] !== "") return;

      cells[index] = currentPlayer;
      e.target.textContent = currentPlayer;

      if (checkWin()) {
        statusDisplay.textContent = `Player ${currentPlayer} wins!`;
        gameActive = false;
        return;
      }

      if (!cells.includes("")) {
        statusDisplay.textContent = "It's a Draw!";
        gameActive = false;
        return;
      }

      currentPlayer = currentPlayer === "X" ? "O" : "X";
      statusDisplay.textContent = `Player ${currentPlayer}'s turn`;
    }

    function checkWin() {
      return winningCombinations.some(combo => {
        const [a, b, c] = combo;
        return cells[a] && cells[a] === cells[b] && cells[a] === cells[c];
      });
    }
    function resetGame() {
      currentPlayer = "X";
      gameActive = true;
      cells = Array(9).fill("");
      statusDisplay.textContent = `Player ${currentPlayer}'s turn`;
      createBoard();
    }
    createBoard();
  </script>
</body>
</html>
```



C:/users/hp/Desktop/tictoc.html

Tic Tac Toe

Player X wins!

Fig 2 Online web-based Game

2.3 Real-World Applications of AI in Daily Life

Application Area	AI Use Case Example	Tool/Technology Used
Smart Assistants	Voice control, reminders, home automation	Alexa, Siri, Google Assistant
Social Media	Face filters, content recommendation, spam control	Instagram AI, Facebook AI
E-commerce	Product suggestions, pricing optimization	Amazon AI, Shopify ML
Transportation	Self-driving cars, traffic predictions	Tesla Autopilot, Google Maps AI
Education	Auto-grading, tutoring bots, plagiarism detection	Gradescope, Grammarly AI
Healthcare	Disease detection from scans, virtual diagnosis	IBM Watson, Zebra Medical
Finance	Fraud detection, credit scoring, chatbots	Zest AI, Upstart, AI chatbots

2.4 Benefits of AI in Daily Life

- **Efficiency**: AI completes tasks faster than humans.
- **Accessibility**: Voice recognition helps users with disabilities.
- **Personalization**: Apps recommend content suited to individual preferences.
- **Cost Saving**: Automates repetitive or labor-intensive tasks.
- **Creativity**: AI assists in art, music, and design without expert skills.

2.5 Challenges to Consider

- **Bias** in AI models due to poor training data.
- **Privacy issues** with voice and image recognition tools.
- **Job displacement** due to automation.
- **Overreliance on AI** leading to reduced human skills.

3. CLIENT-SIDE WEB PROGRAMMING

Client-side web programming is responsible for everything a user sees and interacts with in a browser. It makes web pages **visually appealing**, **responsive**, and **interactive** without needing server-side interactions for every action.

3.1 What is Client-Side Programming?

Client-side programming refers to code that runs **inside the user's browser** rather than on a server. It focuses on:

- User interface (UI) design
- Interactive elements
- Real-time response to user actions

Key Technologies:

- **HTML** – For structure (skeleton)
- **CSS** – For styling (appearance)
- **JavaScript** – For interactivity (behavior)

3.2 CSS (Cascading Style Sheets)

What is CSS?

CSS is a stylesheet language used to describe the look and formatting of a document written in HTML. It separates **content** from **design**, allowing consistent styling across pages.

Key Features of CSS:

- Controls **layout**, **colors**, **fonts**, **animations**, and **responsive design**
- CSS can be:
  - **Inline** (within HTML tag)
  - **Internal** (within `<style>` tags in `<head>`)
  - **External** (in separate `.css` file)

Basic CSS Syntax:

```
selector {
  property: value;
}
```

Examples of CSS Properties:

```
body {
  background-color: #f0f0f0;
  font-family: Arial, sans-serif;
}

h1 {
  color: navy;
  text-align: center;
}

.button {
  padding: 10px 20px;
  border-radius: 10px;
  background-color: #4CAF50;
  color: white;
}
```

Responsive Design (Media Queries):

Make pages look good on all screen sizes:

```
@media (max-width: 768px) {
  body {
    font-size: 14px;
  }
}
```

CSS Animations:

```
@keyframes fadeIn {
  from { opacity: 0; }
  to { opacity: 1; }
}

.fade {
  animation: fadeIn 2s;
}
```

3.3 JavaScript (JS)

What is JavaScript?

JavaScript is a scripting language that enables interactive web features such as:

- Form validation
- Image sliders
- Dynamic content loading
- API communication

It runs **in the browser**, alongside HTML and CSS.

Basic JS Syntax:

```
function greet() {
  alert("Welcome to the website!");
}
```

💡 Key Concepts:

- **Variables**: `let`, `const`, `var`
- **Functions**: Reusable blocks of code
- **Events**: Actions triggered by the user (clicks, hovers)
- **DOM Manipulation**: Change content dynamically

Example: Button Click Alert

```
<button onclick="sayHello()">Click Me</button>

<script>
  function sayHello() {
    alert("Hello, User!");
  }
</script>
```

JavaScript and the DOM

```
document.getElementById("myText").innerHTML = "Changed text!";
```

Integrating with APIs

```
fetch('https://api.example.com/data')
  .then(response => response.json())
  .then(data => console.log(data));
```

3.4 CSS + JavaScript Together = Power

Combining both enables dynamic, stylish interfaces:

Example: Toggle Theme Button

```
<button onclick="toggleMode()">Toggle Dark Mode</button>

<script>
  function toggleMode() {
    document.body.classList.toggle("dark-mode");
  }
</script>

<style>
  .dark-mode {
    background-color: #121212;
    color: white;
  }
</style>
```

Why Learn Client-Side Web Programming?

Skill	Importance
CSS	Custom styling, responsive layouts
JS	Making pages interactive, dynamic behavior
Combined	Full control over user experience on the frontend

Real-Life Use Cases

Application	CSS Usage	JS Usage
Portfolio Website	Layout and fonts	Interactive tabs
E-commerce Site	Product cards styling	Cart functionality
Online Quiz App	Question display	Score calculation
Weather App	Background visuals	Fetching live data

4. BLOCK = DIGITAL NOTEBOOK

To understand blockchain technology, it’s important to first understand what a **“block”** is. A **block** in a blockchain can be visualized as a **page in a digital notebook**, where information is written in a secure, timestamped, and verifiable way. Once written, it cannot be changed—ensuring **immutability and trust**.

4.1 What is a Block?

A **block** is a container of data. It stores:

- **Transaction details or records**
- A **timestamp** (when the block was created)
- A **unique ID** (hash of the block's content)
- A **reference to the previous block** (previous hash)

Just like writing on a new page in a notebook, every new block stores fresh data and links back to the previous block, forming a **chain of blocks**.

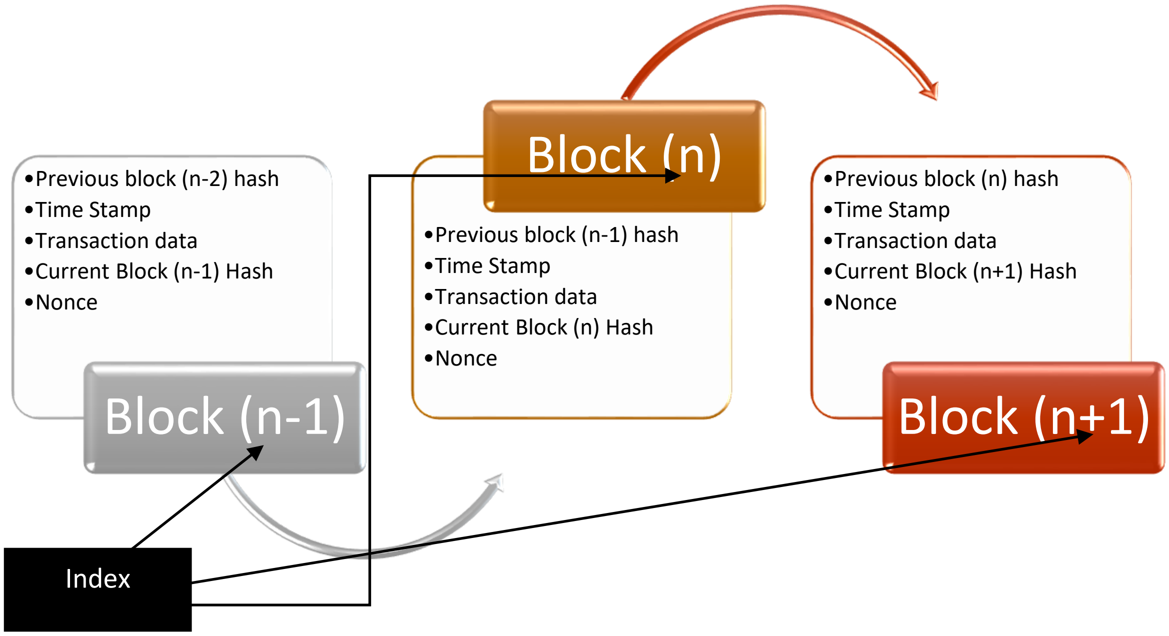


Fig 3 Interconnectivity of Blocks in Block chain

4.2 Structure of a Block

Block Field	Description
Index	Block number in the chain
Timestamp	When the block was created
Data	Transactions, messages, or smart contract instructions
Previous Hash	Digital fingerprint of the previous block
Current Hash	Unique fingerprint of this block (generated using cryptography)
Nonce	Random number used in mining to find a valid hash (Proof-of-Work chains)

Linking blocks:
Each block contains the **hash of the previous block**, making it impossible to tamper with one block without changing all the others.

4.3 Digital Notebook Analogy

Digital Notebook Concept	Blockchain Equivalent
Each page stores notes	Each block stores data
Pages are numbered	Blocks have an index
You can’t change a past page without tearing others	Blocks are immutable
Pages are linked in sequence	Blocks are linked via hashes

This analogy helps learners understand that **just like tampering one page affects the entire notebook**, tampering with one block corrupts the entire blockchain.

4.4 What Type of Data Can Be Stored in a Block?

- **Financial transactions** (e.g., Bitcoin payments)
- **Smart contract execution logs**
- **Health records (in healthcare blockchains)**
- **Educational certificates**
- **Supply chain data**
- **NFT metadata**

Example (Bitcoin block stores):

- Sender address
- Receiver address
- Amount
- Time
- Signature

4.5 Why Blocks Are Secure

- **Hashing**: Each block's content is hashed (converted to a fixed-length code), and even a minor change will generate a completely different hash.
- **Chaining**: Every block is linked to its predecessor. Changing one would require altering the entire chain—extremely difficult in a distributed system.
- **Decentralization**: Multiple copies of the chain are stored across a network, making it hard to manipulate or fake data.

4.6 Real-World Use Cases of Blocks

Industry	Block Contents
Finance	Transaction logs, account updates
Healthcare	Patient records, diagnosis history
Education	Academic transcripts, attendance logs
Supply Chain	Inventory movement, product certifications
Voting	Voter identity hash, voting result

4.7 Summary: Why "Block = Digital Notebook"?

- Just like a notebook records important info sequentially, **blockchains store data in blocks**.
- **Tamper-proof**: A block is write-once, read-many—like a permanent ink entry in a diary.
- **Organized and chronological**: Just like flipping pages, blockchains allow historical tracking of every action or transaction.

5. BLOCKCHAIN

Blockchain is a **decentralized**, **distributed**, and **immutable ledger** that records transactions and data securely. Its design ensures transparency, traceability, and trust—making it useful far beyond just cryptocurrency. A blockchain is made up of a series of **linked blocks**, each containing verified data. Once data is stored, it **cannot be altered** without consensus from the network.

5.1 Role of Blockchain in Various Sectors

Blockchain is revolutionizing how industries handle data, transactions, and trust. Below are some of its most important real-world applications:

💹 A. Finance

- **Use Cases**:
    - Cryptocurrency (Bitcoin, Ethereum)
    - Peer-to-peer payments (e.g., Lightning Network)
    - Smart contracts for lending/borrowing
    - Tokenized assets (stocks, real estate)
- **Benefits**:
    - Lower transaction fees
    - Faster cross-border payments
    - Transparent auditing

⛨ B. Healthcare

- **Use Cases**:
    - Secure sharing of patient records between hospitals
    - Prescription drug tracking
    - Preventing medical insurance fraud
- **Benefits**:
    - Enhanced patient privacy and control
    - Accurate medical histories
    - Tamper-proof clinical trials data

🎓 C. Education

- **Use Cases**:
    - Issuing verifiable digital degrees/certificates
    - Recording student achievements
    - Preventing falsification of credentials
- **Benefits**:
    - Eliminates fake degrees
    - Easy verification by employers
    - Transparent academic history

5.2 DeFi (Decentralized Finance) vs CeFi (Centralized Finance)

Feature	DeFi (Decentralized)	CeFi (Centralized)
Control	User controls funds via private wallet	Third-party (exchange) manages funds
Trust	Trustless – relies on smart contracts	Trust in company/platform
Accessibility	Permissionless – anyone can join	Often requires KYC/approval
Transparency	Fully open and auditable code	Limited insight into operations
Examples	Uniswap, Aave, Compound	Coinbase, Binance, PayPal

DeFi allows users to **borrow, lend, swap, and earn** without banks or intermediaries.

5.3 DEX vs CEX

What Is a CEX (Centralized Exchange)?

A **Centralized Exchange** is a platform operated by a company that acts as a **middleman** between buyers and sellers. Examples: **Binance, Coinbase, Kraken**.

✅ **Features:**

- User accounts with KYC/AML verification
- Order matching controlled by the company
- Funds are held in exchange-controlled wallets

⚠ **Pros:**

- High liquidity
- Faster transactions
- Customer support available
- Advanced trading tools

❌ **Cons:**

- Risk of hacking
- Custodial: you don’t own the private keys
- Regulatory dependencies

What Is a DEX (Decentralized Exchange)?

A **Decentralized Exchange** allows **peer-to-peer** crypto trading without intermediaries. Examples: **Uniswap, PancakeSwap, SushiSwap**. **Icecreamswap, Shadowswap**

✅ **Features:**

- Runs on blockchain via smart contracts
- Users trade directly from their wallets
- No KYC (in most cases)

⚠ **Pros:**

- Non-custodial: users keep control of funds
- Greater privacy and anonymity
- Permissionless and open source

❌ **Cons:**

- Lower liquidity (in some cases)
- Higher slippage on large trades
- Slower trade execution
- No direct customer support

DEX vs CEX Comparison Table

Feature	CEX	DEX
Control	Centralized (company-controlled)	Decentralized (smart contracts)
Custody	Custodial (exchange holds your crypto)	Non-custodial (you hold your crypto)
KYC/AML	Required	Often not required
Security Risk	Higher (target for hackers)	Lower (users control private keys)
Liquidity	High	Variable (depends on the pool)
Speed	Fast (centralized servers)	Slower (blockchain-dependent)
Fees	Trading + withdrawal fees	Mostly network (gas) + swap fees
User Interface	Beginner-friendly	Sometimes technical/complex
Customer Support	Yes	No

5.4 Blockchain Supported by BlackDoge

BlackDoge is a **meme-inspired blockchain platform** with functional support for smart contracts, tokens, NFTs, and DeFi apps. It merges community engagement with technical features.

✅ **Key Features:**

- **Smart Contract Compatibility** (EVM-based)
- **Fast block confirmation times**
- **Low gas fees**
- **Community governance**
- Supports **NFT minting**, **decentralized apps (dApps)**, and **token creation**

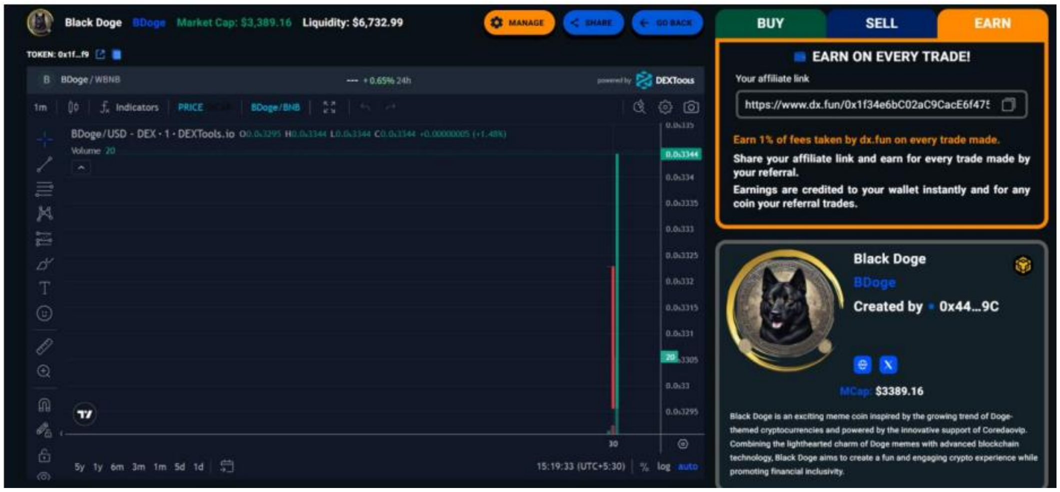
🔍 Use Cases:

- Meme token trading
- Crowdfunding via smart contracts
- NFT marketplaces
- Play-to-Earn gaming

Blockchain
1. ERC Chain Supply: 1,000,000,000 BDoge
2. Core Chain Supply: 1,000,000,000 BDoge
3. BEP20 Chain Supply: 1,000,000,000 BDoge
4. Polygon Chain Supply: 1,000,000,000 BDoge
5. ARB Chain Supply: 1,000,000,000 BDoge
6. Base Chain Supply: 1,000,000,000 BDoge
7. Fantom Chain supply: 1,000,000,000 BDoge
8. Omax Chain supply: 1,000,000,000 BDoge
9. Pulse Chain Supply: 1,000,000,000 BDoge
10. Celo Chain Supply: 1,000,000,000 BDoge

How to Get Black Doge on BEP 20 Chain (Fun pool)

1. Visit https://www.dx.fun/0x1f34e6bC02aC9CacE6f47562815b17294a227Ff9?affiliate=0x4470a588d84Aed1998bB3007F198B82694175F9C&chainID=56
2. Set up a metamask wallet compatible with the BEP20 chain.
3. Make Buy, Sell transaction



Black Doge BDoge Market Cap: $3,389.16 Liquidity: $6,732.99

TOKEN: 0x1f...f9

BDoge / WBNB --- +0.65% 24h

BDoge/USD - DEX · 1 · DEXTools.io

Volume 20

EARN ON EVERY TRADE!

Your affiliate link: https://www.dx.fun/0x1f34e6bC02aC9CacE6f475

Earn 1% of fees taken by dx.fun on every trade made.

Share your affiliate link and earn for every trade made by your referral.

Earnings are credited to your wallet instantly and for any coin your referral trades.

Black Doge

BDoge

Created by 0x44...9C

MCap: $3389.16

Black Doge is an exciting meme coin inspired by the growing trend of Doge-themed cryptocurrencies and powered by the innovative support of Coredaovip. Combining the lighthearted charm of Doge memes with advanced blockchain technology, Black Doge aims to create a fun and engaging crypto experience while promoting financial inclusivity.

Fig 4 Working with Blackdoge Funpool

Ethereum chain

https://www.dx.fun/0xAA958B65bc7AdbE23F35fcD9c8E6d7FeA0C1AbdE?affiliate=0x4470a588d84Aed1998bB3007F198B82694175F9C&chainID=1

Arbitrum chain

https://www.dx.fun/0x58c3dC7e8A0fD1d17b4d91864E1535A82Ec6d014?affiliate=0x4470a588d84Aed1998bB3007F198B82694175F9C&chainID=42161

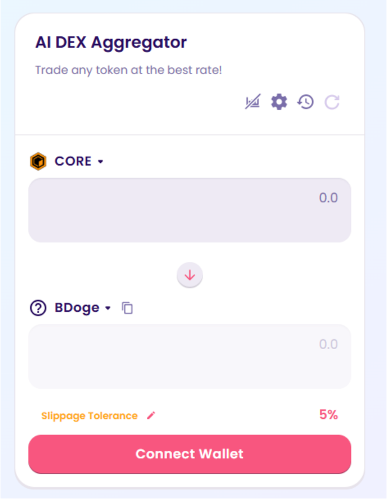
Base Chain

https://www.dx.fun/0x358994fdc7f192a894Bf2468e6FD09DD8507f294?affiliate=0x4470a588d84Aed1998bB3007F198B82694175F9C&chainID=8453

Accessing BlackDoge by DEX (Icecream swap, SushiSwap)

Core chain

https://icecreamswap.com/swap?outputCurrency=0x246aFA57E85fD92e7CF0efEb4Bf20d422FC4F0AB



AI DEX Aggregator

Trade any token at the best rate!

CORE

0.0

BDoge

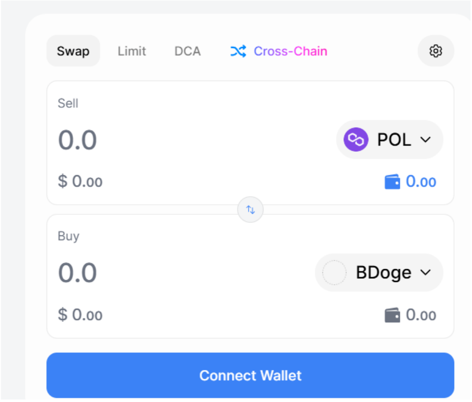
0.0

Slippage Tolerance 5%

Fig 5 Buying Blackdoge on Icecreamswap

Polygon chain

https://www.sushi.com/polygon/swap?token0=NATIVE&token1=0x5964f01D14C344BD3bc8BEb2554b3AC66CD6b755



Sell
0.0 POL
$ 0.00 0.00

Buy
0.0 BDoge
$ 0.00 0.00

Fig 6 Buying Blackdoge on Icecreamswap

✓ Summary

Sector	Blockchain Role	Key Benefit
Finance	Secure, fast, low-cost transactions	Financial freedom, DeFi access
Healthcare	Record management, anti-fraud systems	Privacy, data integrity
Education	Digital certificates, learning records	Verification, transparency
BlackDoge	Blockchain ecosystem for DeFi, NFT, dApps	Innovation and decentralization

6. COREDAOVIP TOKEN SUMMARY & WHITE PAPER HIGHLIGHTS

◆ Token Basics (from the white paper)

- **Token Name**: COREDAOVIP (Ticker: COREVIP)
- **Total Supply**: **100,000 tokens** (fixed with no additional minting)
- **Built on**: Core blockchain, powered by 9NFTMANIA, influencing VIP NFT projects on YoungParrot marketplace
- **Official website:** https://coredao.vip
- **Visit white paper** https://coredao.vip/coredaovip.pdf

Token Mechanics & Circulation

- **Liquidity Integration**: When CoreDaoVIP is purchased, **50% of the CORE token proceeds are automatically converted to USDT**, which helps back USDT liquidity pools for price stability. Price automation occurs via Core–USDT swaps based on CORE price fluctuations.
- **Distribution Mechanism**: Primarily distributed via **airdrops to 9NFTMANIA NFT holders**. Tokens sold by users are automatically repurchased into liquidity pools, ensuring controlled circulation and price consistency.
- **New Pools**: Periodic creation of pools pairing CoreDaoVIP with CORE, USDT, YPC, SHIB, DOGE, etc., encouraging wider ecosystem engagement and liquidity expansion.

Use Cases: DeFi, Liquidity Pools & Auto-Trading

Based on peer-reviewed analysis:

- **Liquidity Pool Utility**: CoreDaoVIP enables deeper liquidity across several DEXs—IceCreamSwap, LFGSwap, ShadowSwap, ArcherSwap—reducing slippage and supporting smoother trades
- **Automated Trading**: Supports algorithmic trading, AMM strategies, and yield farming—benefiting liquidity providers through fee revenue and trading efficiency.
- **Profit Mechanism for LPs**: Through fee sharing and staking-like incentives, LPs may earn rewards similar to traditional brokers but with decentralized transparency

Governance & Strategic Vision

- Token holders are intended to partake in governance decisions, such as fee structures and protocol direction, aligning with the CoreDaoVIP ethos of decentralized control
- The project emphasizes a **"Silent Growth Philosophy"**—a deliberate, knowledge-first approach to building decentralized systems, rather than chasing hype or speculation

Tokenomics Snapshot

Feature	Details
Total Supply	100,000 tokens (fixed)
Circulation Model	Airdrops to NFT holders; buybacks via liquidity pools
Liquidity Strategy	CoreDaoVIP + CORE/USDT pools with automated balancing
Main Utility	Governance, DeFi access, liquidity enhancement
Supported DEXs	IceCreamSwap, LFGSwap, ShadowSwap, ArcherSwap
Hold Return Model	Passive LP rewards, token appreciation, DeFi incentives
Governance Structure	Token-based decision-making by holders
Philosophy	Long-term, ethical value creation over transient cycles

Why This Matters

- **Scarcity Value**: With only 100,000 total tokens, even small upticks in demand can significantly move the market.
- **Automated Price Management**: Liquidity pools dynamically regulate token supply and price by absorbing sales and balancing assets.
- **Dual Utility**: CoreDaoVIP bridges **NFT branding (via 9NFTMANIA and YoungParrot)** with **DeFi mechanics**, offering both collectibility and financial utility.

7. NFT (NON-FUNGIBLE TOKEN)

7.1 Introduction to NFTs

NFTs (Non-Fungible Tokens) are unique digital assets stored on a blockchain that represent **ownership or proof of authenticity** of a digital or physical item. Unlike cryptocurrencies like Bitcoin or Ethereum, which are **fungible** (1 BTC = 1 BTC), NFTs are **one-of-a-kind**, with no identical replacement.

✓ Key Properties of NFTs

- **Uniqueness**: Each NFT has a distinct identifier.
- **Ownership**: Verifiable on-chain proof of who owns the asset.
- **Indivisibility**: NFTs cannot be split like cryptocurrencies.
- **Transferability**: Easily sold, traded, or gifted on marketplaces.
- **Metadata**: Stores information such as artist name, title, creation date, royalty settings, and more.

7.2 International NFT Marketplaces

A. OpenSea

- **Overview**: One of the largest and most well-known NFT marketplaces.
- **Blockchain Support**: Ethereum, Polygon, Solana, Base
- **Asset Types**: Art, domain names, music, trading cards, virtual worlds
- **Features**:
  - Creator royalties
  - Bidding and auctions
  - Wallet support (MetaMask, WalletConnect)
- **Website**: https://opensea.io

B. YoungParrot

- **Overview**: A rising NFT marketplace built on **Core Blockchain**.
- **Unique Selling Point**:
  - Focuses on **community-driven NFTs**
  - Seamless integration with **9NFTMANIA**
- **Use Cases**:
  - VIP NFT minting and trading
  - Access to CoreDaoVIP token airdrops
- **Supported Wallets**: Core wallet, MetaMask (configured for Core chain)
- **Website**: https://youngparrot.com

7.3 NFT Brand: 9NFTMANIA

What is 9NFTMANIA?

9NFTMANIA is a prominent **NFT brand and ecosystem** that:

- Mints exclusive, rare NFTs
- Grants utility in **token airdrops**, **DeFi**, and **membership perks**
- Is closely integrated with **CoreDaoVIP** and **YoungParrot**
- https://9nftmania.com

Key Features:

Feature	Description
NFT Utility	Holding a 9NFTMANIA NFT can qualify you for **CoreDaoVIP airdrops**
Brand Identity	Unique art collections, rarity scores, and creative themes
Community Role	NFTs may represent access, loyalty, or voting rights in decentralized platforms
Reward Mechanism	NFT holders receive early access to pools, DeFi tools, and token drops

7.4 Significance of NFTs

In Digital Art:

- Artists gain control over royalties
- Art can't be faked or stolen easily
- Worldwide visibility for creators

In Gaming:

- In-game assets like weapons, skins, and avatars are tokenized and tradable
- Players truly **own** their achievements

In Events and Access:

- NFTs as **tickets** or **membership cards** (e.g., concerts, clubs, VIP lounges)

In DeFi:

- NFTs as **collateral** for loans
- NFT-based **yield farming** and governance

In Education:

- NFT certificates and badges for online courses (immutable, verifiable)