

 Store with IPFS – Decentralized File Upload  
  
**Objective/Aim:**  
  
 To learn how to upload and access files using IPFS (InterPlanetary File System), understand decentralized storage, and observe how content-addressed data is shared without central servers.

**Apparatus/Software Used:**

* Laptop/PC
* IPFS (via https://web3.storage or https://infura.io)
* Browser or command-line IPFS client
* Internet

**Theory/Concept:**

**What is IPFS?**

IPFS is a **peer-to-peer**, **decentralized file system** that allows users to store and share files via **content-addressing** (using CIDs – Content Identifiers).

**Key Concepts:**

* **CID (Content Identifier):** Unique hash generated from file content.
* **Decentralized Storage:** Files are stored across many nodes, not on a single server.
* **Persistence:** Files must be pinned to stay on the network (via pinning services like Pinata or web3.storage).

**How it works:**

* Files are broken into chunks and hashed.
* Each chunk has a unique CID.
* Files can be retrieved using their CID from any IPFS node or public gateway (like https://ipfs.io/ipfs/<CID>).



**Procedure:**

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

1. Visit https://web3.storage and create a free account.
2. Upload any file (e.g., .txt, .png, .pdf) via the dashboard.
3. After upload, copy the generated CID.
4. Access your file using a gateway URL like:  
   https://ipfs.io/ipfs/<your\_CID>
5. Test retrieving the file from different browsers and devices.



**Observation Table:**

| **Feature** | **IPFS** |
| --- | --- |
| Definition | A peer-to-peer decentralized file system using content addressing |
| Control | Controlled by users across a distributed network |
| Data Ownership | Users own and manage their own data |
| Examples | IPFS, Filecoin, web3.storage, Pinata |
| Privacy | High privacy; data is not monetized by centralized platforms |
| Accessibility | Accessible via CIDs and public IPFS gateways |
| Security | Secured using cryptographic hashing and distributed design |
| Censorship | Resistant to censorship due to its decentralized nature |
| Scalability | Faces some scalability challenges but is evolving with new protocols |



