

## 1. IPFS

# IPFS

IPFS – the Inter-Planetary File System – is a decentralized file storage system, that allows anyone to join the network and host content. IPFS works a lot like peer-to-peer filesharing services (like bittorrent), but one key difference is that on IPFS content is addressed by its hash (as opposed to a filename).

To get your data onto IPFS, you can:

1. **Use a third-party node** – to keep data online permanently you can “pin” it to a node. Several companies provide “pinning” services that allow you to upload files to their IPFS nodes and pin the files there (so they are not deleted from the node).

2. **Run your own IPFS node to host your data** – this will make your data available as long as your node is running.

1. When another IPFS node downloads a piece of data, it typically caches it, making your data available on another node as well. Individual node providers can choose whether they want to replicate your data (i.e., cache it locally), but as with bittorrent, you would expect that popular files are highly replicated, whereas unpopular files can easily disappear off of the network if their source node goes offline.

## (Free) Pinning Services

For this assignment you will need to upload files to IPFS. You can install your own IPFS node, or you can use a (free) provider to store a small number of files.

## Infura

You can upload files to Infura’s pinning service (100Mb limit). Infura provides an example in Python that is essentially all you need.



The old instructions suggest using “get” to retrieve data from IPFS.  
The new documentation suggests using “cat”.

## 1. IPFS

### Pinata

---

Pinata provides free IPFS pinning services, but before you can use it, you'll need to sign up for a (free) account at [Pinata](#). Pinata also has its own "[docs](#)" pages that include tutorials.

### IPFS Gateways

Once data is stored on IPFS, it can be discovered and accessed by IPFS nodes. If you want to access it using http (i.e., without running an IPFS node), you will need to get it from a "gateway." A gateway is simply an IPFS node that allows access to IPFS through traditional http requests.

To access data through a gateway, you will only need its "content ID" (CID). For example, the metadata (and image) for Beeple's Everydays (which [sold for \\$69M at Christies](#)) is on IPFS, with CID [QmPAG1mjxcEQPPtqsLoEcauVedaeMH81WXDPvPx3VC5zUz](https://ipfs.io/ipfs/QmPAG1mjxcEQPPtqsLoEcauVedaeMH81WXDPvPx3VC5zUz)

### Infura

---



```
https://ipfs.infura.io:5001/api/v0/cat?arg={content ID}
```



Note that Infura only supports POST requests (not GET), so you cannot directly copy that URL into your browser's address bar; you need to use a library (like Python requests) that allows POST requests

### Pinata

---

The pinning service, Pinata, provides an [IPFS gateway](#).

## 1. IPFS

### Protocol Labs

---

Protocol Labs (the creators of IPFS) provides a free gateway.



```
https://ipfs.io/ipfs/{content ID}
```

### Cloudflare

---

Cloudflare provides an IPFS gateway.



```
https://cloudflare-ipfs.com/ipfs/{content ID}
```

## Assignment

Using the template provided at [ipfs.py](https://ipfs.py), create two functions

1. "pin\_to\_ipfs()" which takes a Python dictionary, and stores the dictionary (as JSON) on IPFS. The function should return the Content Identifier (CID) of the data stored.
2. "get\_from\_ipfs(cid)" which takes as input an IPFS CID and returns a Python dictionary containing the content. For this function, you may assume the content identified by the CID is valid JSON content. For example, you may assume that the CID does not refer to an image, video or other type of binary content that cannot be easily JSONified.

You can get feedback on your code by clicking "Run Test Cases".

Note that the test cases use a fixed data point (bored ape #489) to test your code performance against existing pinned data. The Auto-grader uses a random "bored ape" from the collection queried during grading. We recommend you perform your own tests against several different IPFS pins from the collection (which can be found here:

<https://gateway.moralisipfs.com/ipfs/QmeSjSinHpPnmXmspMjwiXyN6zS4E9zccariGR3jxcaWt> before submitting to the final grading script. Any errors identified by the test cases

## 1. IPFS

over time. It may work for some open files and not the other files that the final grader will use.

When you are happy with it, click “Education->Mark as Completed” to submit. Once you mark the unit as completed, you cannot re-submit, so make sure you’re happy with everything before you submit!

Next ▶