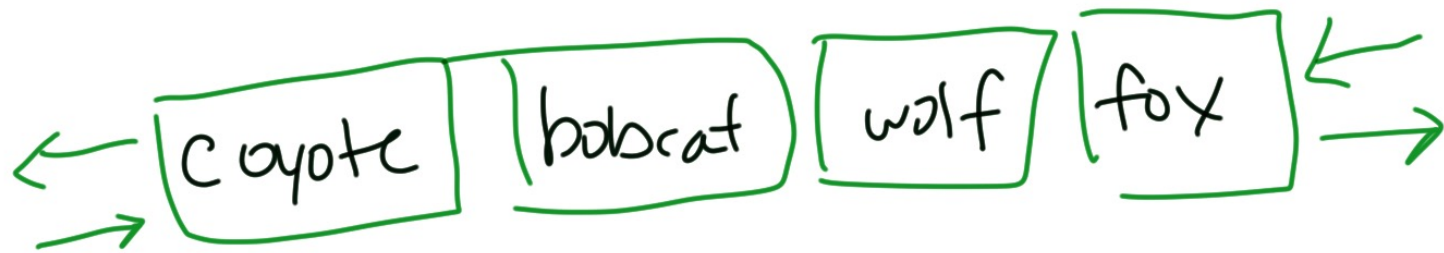


Deque "Deck"

→ add + remove from both ends

→ combo of stack + queue
(LIFO) (FIFO)



* think of keyboard stream

* web browser history

- <https://visualgo.net/en/list>
- Queue then deque

WS. add to front (news)
WS. remove from front()
WS. get front()

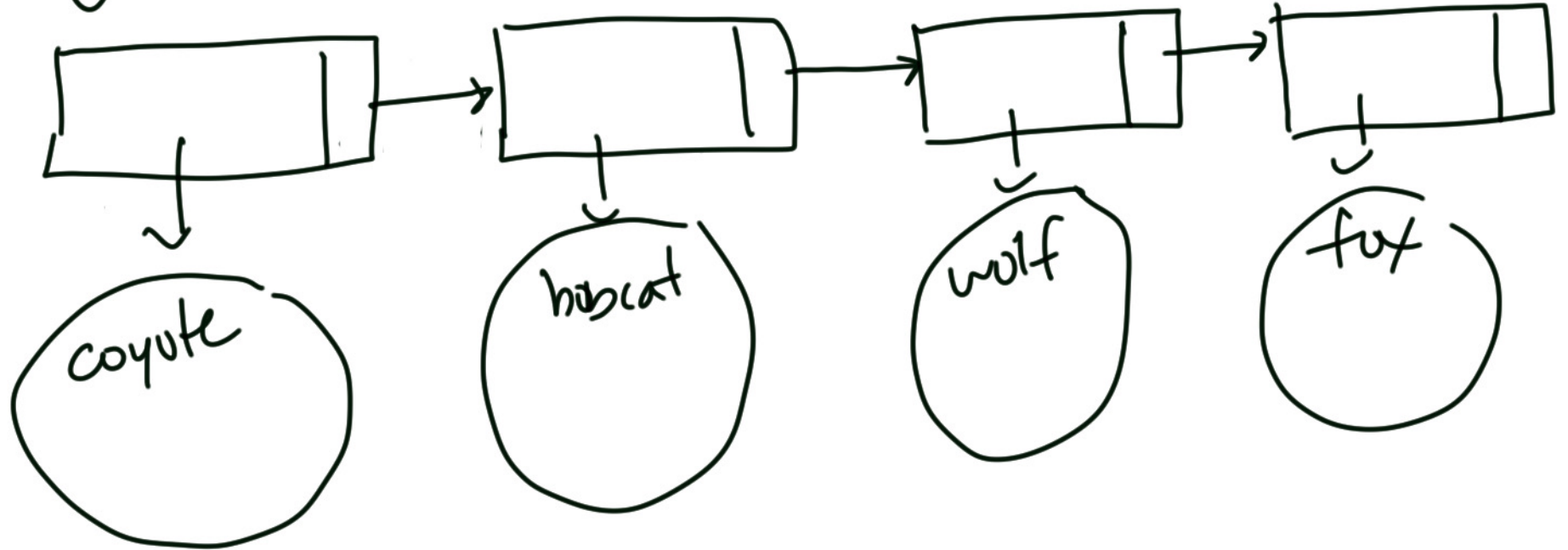
WS. add to back (weather)
WS. remove from back()
WS. get back()



Queue WS of
websites

Deque Implementation

first Node
↓

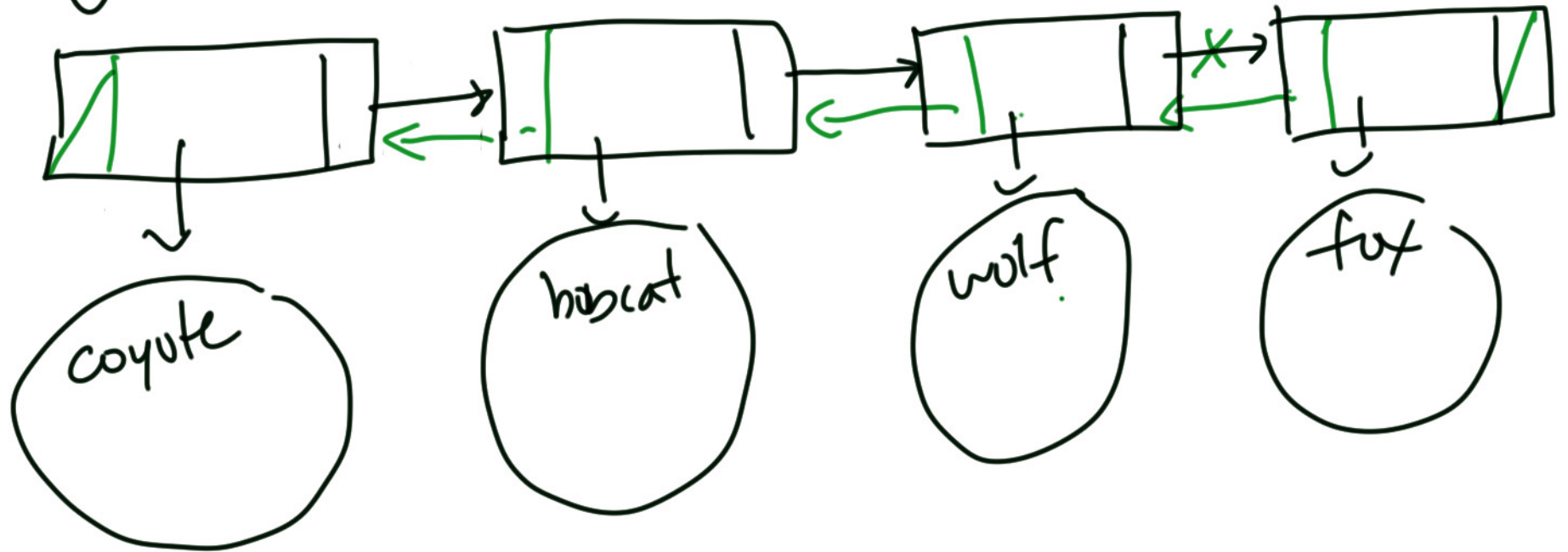


Deque Implementation

firstNode
↓

USE doubly linked node

lastNode
↓



Doubly Linked Deque

Notice there is a firstNode and lastNode reference

```
11 public class LinkedDeque<T> implements DequeInterface<T> {
12
13     private DLNode firstNode;
14     private DLNode lastNode;
15
16     public void initialize() {
17         firstNode = null;
18         lastNode = null;
19     }
20
21     .....
```

Nested Doubly Linked Node

Notice the node has a previous, data and next field!

```
.....
113 private class DLNode
114 {
115     private T data; // Entry in stack
116     private DLNode next; // Link to next node
117     private DLNode prev; // Link to previous node
118
119     private DLNode(T dataPortion)
120     {
121         this(null, dataPortion, null);
122     } // end constructor
123
124     private DLNode(DLNode prevPortion,
125                   T dataPortion,
126                   DLNode nextPortion
127                   )
128     {
129         prev = prevPortion;
130         data = dataPortion;
131         next = nextPortion;
132
133     } // end constructor
134
.....
```

```
@Override
```

```
public void addToBack(T newEntry) {  
    DLNode newNode = new DLNode(lastNode, newEntry, null);  
  
    if (isEmpty()) {  
        firstNode = newNode;  
    } else {  
        lastNode.setNext(newNode);  
    }  
  
    lastNode = newNode;  
}
```


Deque addToBack

