

Intro to Linked Lists

Learn to Code with Rust

Recursive Data Structures

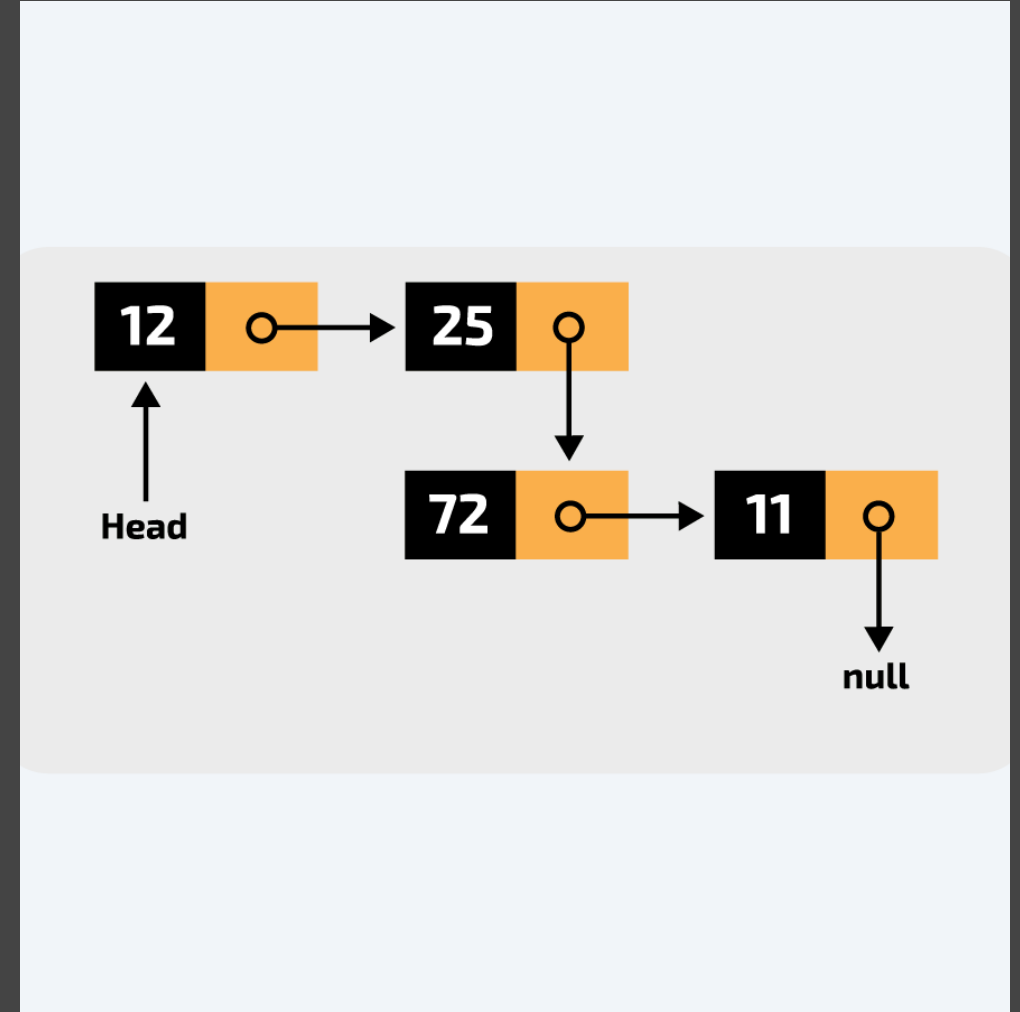
- **Recursion** is when a function calls itself.
- A **recursive data structure** is one that stores the structure itself.
- Imagine a struct with a field whose type is that exact struct.

Recursive Data Structures

- Recursive functions use a **base case** to force the termination of recursion.
- Recursive data structures use a **Box** to solve the infinity problem.
- The compiler does not have to worry about nested data structures occupying infinite memory.




Linked List

- A **linked list** is a data structure that consists of a linear collection of connected nodes.
- Each node stores a piece of data and a reference to the next node in sequence.
- Unlike arrays/vectors, the nodes are not stored contiguously in memory which makes it easier to insert an item in the middle of the list.





Q Custom order ≡

#	Title	Album	Date added	
1	 Complicated ↓ Avril Lavigne	Let Go	1 second ago	4:05
2	 Sk8er Boi ↓ Avril Lavigne	Let Go	1 second ago	3:24
3	 I'm with You ↓ Avril Lavigne	Let Go	1 second ago	3:43

How Do We Model This?

```
struct MusicPlaylistItem {  
    name: String,  
    artist: String,  
    nextTrack: MusicPlaylistItem  
}
```