# Lists in Prolog

Prolog lists may be heterogeneous: They may contain elements of different „types“

- Example: Homogeneous lists

| | |
|---|---|
| `[1, 2, 3]` | List of integers |
| `['a', 'b', 'c']` | List of characters |
| `[ ]` | Empty list |
| `[[1,2], [ ], [5]]` | List of lists |

- Example: Homogeneous only at the top level

| | |
|---|---|
| `[[1,2], [ ], ['a']]` | List of lists but the element types differ |

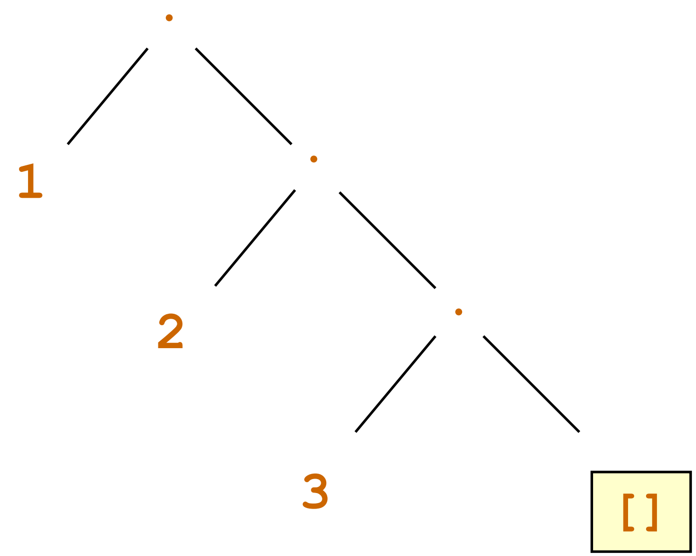
- Example: Fully heterogeneous

```
[[1,2], 'a', 3]
```

# List are Binary Trees Encoded as Terms (1)

- Internally, lists are binary trees whose leaves are the lists elements:

`[1, 2, 3]` =

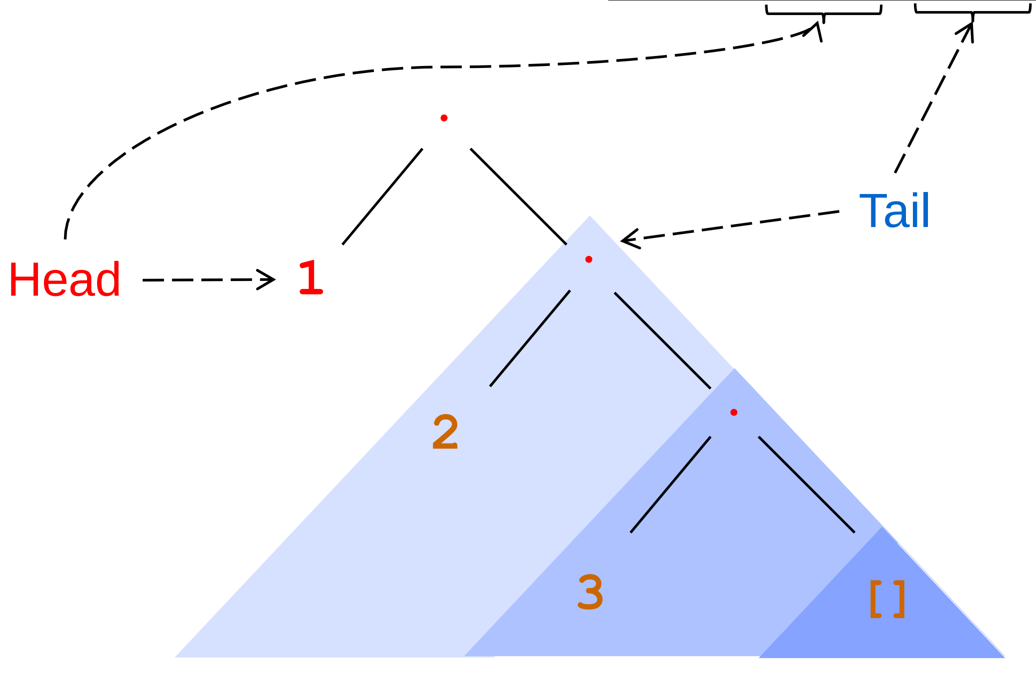



Each list is terminated by an empty list

# List are Binary Trees Encoded as Terms (2)

- The functor '.' is the list constructor:

```
?- .(  1  ,[2,3]) = [1, 2, 3]
```




- The first element is the „head“ the second is the „tail“.
- The „tail“ is the list of all the other elements.

# Accessing Head and Tail

- Notation [ Head | Tail ]

```
?- [1,2,3,4]=[H|T].              ------> H=1,  T=[2,3,4]

?- [1,2,3,4]=[H1,H2|T].          ------> H1=1,  H2=2,  T=[3,4]

?- [1,2,3,4]=[_,_,H|_].          ------> H=3

?- [1,2,3,4]=[_,_,_,_|T].        ------> ???

?- []=[H|T].                     ------> ???

?- [1,2,3,4]=[_,_,_,_,_|T].      ------> ???
```

```
?- X = [Y,2,3,4], Y=1.           ------> X=[1,2,3,4],  Y=1

?- T = [2,3,4], X=[1|T].         ------> ???
```