

اتولیسپ : درس هفتم

نویسنده : عبدالحکیم قدس

در این بخش می پردازیم به توابع کار با لیست. همانطور که قبلا اشاره شد لیستها یک از مهمترین مباحث اتولیسپ می باشد که در بسیاری از مواقع به آنها نیازمند فوایم شد
ابتدا بیایید یک لیست بسازیم

```
(list "abc" "12" "xyz")
```

حال لیست بالا را به یک متغیر نسبت می دهیم

```
(setq myList (list "abc" "12" "xyz"))
```

برای بدست آوردن اولین مقدار یک لیست از تابع `car` استفاده می شود به مثالها و نتایج بازگشتی توجه کنید.

```
(car (list 3 5 7))          3  
(car myList)              "abc"
```

همانطور که ملاحظه می کنید تفاوت پندانی بین یک لیست و آرایه دوبردی وجود ندارد
برای بدست آوردن همه ی اعضای لیست به غیر از مقدار اول از تابع `cdr` استفاده می شود

```
(cdr (list 3 5 7 9))      (5 7 9)  
(cdr myList)              ("12" "xyz")
```

دومین عضو لیست را به روش زیر بدست می آوریم

```
(cadr (list 3 5 7 9))     5  
(cadr myList)            "12"
```

توجه داشته باشید در مثال بالا "۱۲" یک رشته ممسوب می شود. و نیز شما می توانید لیستهای ترکیبی از اعداد حقیقی ، صمیع و مقادیر رشته ای داشته باشید

```
(list "ghods" 17.36 "center" 12)
```

برای بدست آوردن سومین عضو یک لیست از `caddr` استفاده می شود

```
(caddr myList)           "xyz"
```

توجه داشته باشید همه ی توابع بالا کاربردهای فراوانی دارند. شاید در ابتدا کمی گیج کننده باشد اما یکی از مهمترین مسائلی که باید در نظر بگیرید اینست که بیشتر داده های اتولیسپ لیستهای سه عضوی هستند. از همینجا اهمیت و کاربرد این توابع هویدا می شود

لیستها می توانند تو در تو هم تعریف شوند. بازهم برمی گردیم به مفهوم آرایه ها

```
(list (list "a" "b" "c") (list 3 5 7))
```

شاید تعجب کنید اما برای بدست آوردن اولین عضو از اولین لیست زیر عضو یک لیست هم تابعی تعریف شده است

```
(caar (list (list "a" "b" "c") (list 3 5 7))) "a"
```

یکی از جالبترین توابع کار با لیست ها **foreach** می باشد. این تابع بر روی هر یک از اعضای لیست یک عمل فواید را تکرار می کند. چیزی شبیه ملقه ها ، البته نه به آن بست و گسترش ولی تابع سرراستی است و کاربردهای فراوانی دارد. بهترینش شاید این باشد که نیازتان به ملقه سازی کمتر فواید شد. نمونه

```
(foreach a mylist (princ a)) abc12xyz"xyz"
```

در فرمان بالا تابع **foreach** هر یک از گزینه های موجود در لیست را به ترتیب به متغیر **a** نسبت می دهد و در ادامه تابع **princ** آنها را چاپ می کند چیزی زیادی به پایان این بخش نمانده است . در ادامه برای افزودن یک عضو جدید به ابتدای لیست به شکل زیر عمل می شود

```
(cons 3 (list 7 5 9)) (3 7 5 9)
(cons "pqr" mylist) ("pqr" "abc" "12" "xyz")
```

البته برای افتمصاص یافتن این عضو جدید به متغیر **myList** باید از **setq** استفاده کنید. اگرچه تنها کاری که شما در بخش قبل انجام دادید نمایش یک سری داده پشت سر هم بود

```
(setq mylist (cons "pqr" mylist))
```

و نیز برای بدست آوردن نامین عضو یک لیست از تابع **nth** استفاده می شود

```
(nth 1 mylist) "abc"
(nth 0 mylist) "pqr"
```

تمام لیستها بدون استثنا از مقدار صفر شروع می شوند. همیشه آرایه صفرم اولین مقدار یک لیست است. این موضوع انتقابی نبوده و درون اتولیسپ یک قانون بشمار می آید امیدوارم مطالب ارائه شده مورد توجه شما قرار گرفته باشد