

برنامه نویسی به زبان

پرل

مرجع کامل

تالیف

مهندس هادی کیامرثی

تمام مثال های موجود در این کتاب با کامپیوتر تست شده اند تا از هر گونه خطا
مبرا باشند با این حال ممکن است باز هم خطاهایی در آن وجود داشته باشد از
کلیه خوانندگان این کتاب ، اساتید و دانشجویان محترم خواهشمندم برای مطلع
کردن مولف از این خطا ها لطفا با ایمیل آدرس زیر تماس بگیرند

hadikiamarsi@gmail.com

لازم به ذکر است کلیه حقوق مادی و معنوی این اثر برای مولف محفوظ می
باشد و هرگونه کپی برداری و استفاده از محتویات این کتاب به هر نوعی تحت
پیگرد قانونی قرار می گیرد

کتابخانه مدرسی

فصل پنجم

فادی پیا مدتی

در این فصل مطالب زیر را خواهید آموخت

ایجاد آرایه

دسترسی به عناصر آرایه

ایجاد آرایه ای از اعداد مرتب

اندازه آرایه

حذف و اضافه آیتم ها در آرایه

استخراج آیتم ها از آرایه

تغییر آیتم های آرایه

تبدیل رشته به آرایه

تبدیل آرایه به رشته

مرتب کردن آرایه ها

تغییر شماره اندیس ابتدایی آرایه

ادغام آرایه ها

انتخاب آیتم ها از لیست ها

کیا مدتی

یک آرایه لیستی از تعدادی متغیر اسکالر می باشد که برای تعریف آن از علامت (@) استفاده می شود . برای دستیابی به یک عنصر از یک آرایه باید اسم آرایه را با علامت (\$) بکار ببرید و در جلوی آن علامت براکت به همراه شماره اندیس آن عنصر قرار دهید .

در زیر مثالی از نحوه استفاده از آرایه ها آورده شده است

```
#!/usr/bin/perl

@ages = (25, 30, 40);
@names = ("John Paul", "Lisa", "Kumar");

print "\$ages[0] = $ages[0]\n";
print "\$ages[1] = $ages[1]\n";
print "\$ages[2] = $ages[2]\n";
print "\$names[0] = $names[0]\n";
print "\$names[1] = $names[1]\n";
print "\$names[2] = $names[2]\n";
```

اجرای کد بالا نتیجه زیر را ظاهر خواهد نمود

```
$ages[0] = 25
$ages[1] = 30
$ages[2] = 40
$names[0] = John Paul
$names[1] = Lisa
$names[2] = Kumar
```

در پرل لیست ها و آرایه ها بسیار شبیه هم هستند با این تفاوت که آرایه ها متغیر هستند ولی لیست ها فقط داده هستند .

ایجاد آرایه

برای ایجاد یک آرایه از علامت @ در ابتدای نام آرایه استفاده نمایید و سپس علامت تساوی قرار می گیرد و در جلوی آن علامت پرانتز یا qw قرار می گیرد و سپس آیتم های آرایه آورده می شوند . به مثال زیر توجه نمایید .

```
@array = (1, 2, 'Hello');
@array = qw/This is an array/;
```

در خط دوم مثال بالا آیتم های آرایه با علامت qw// تعریف شده اند . در مثال بالا آیتم های آرایه با علامت جای خالی از هم جدا شده اند اولین آیتم آرایه کلمه This و چهارمین آیتم آرایه کلمه array می باشد ولی باید بدانید که بجای جدا کننده آیتم جای خالی از جدا کننده خط جدید نیز مانند مثال زیر می توانید استفاده نمایید

```
@days = qw/Monday
Tuesday
...
```

```
Sunday/;
```

شما می توانید از روش زیر نیز برای مقدر اولیه دادن به آرایه نیز استفاده نمایید

```
$array[0] = 'Monday';  
...  
$array[6] = 'Sunday';
```

دسترسی به عناصر آرایه

برای دسترسی به یک آیتم خاص در یک آرایه ابتدا باید نام آرایه را به همراه علامت \$ در ابتدای آن نوشته سپس علامت برکت باز و بسته گذاشته می شود به همراه شماره اندیس آن آیتم . برای روشن شدن موضوع به مثال زیر توجه نمایید

```
#!/usr/bin/perl  
  
@days = qw/Mon Tue Wed Thu Fri Sat Sun/;  
  
print "$days[0]\n";  
print "$days[1]\n";  
print "$days[2]\n";  
print "$days[6]\n";  
print "$days[-1]\n";  
print "$days[-7]\n";
```

اجرای کد بالا نتیجه زیر را ظاهر خواهد نمود

```
Mon  
Tue  
Wed  
Sun  
Sun  
Mon
```

بیاد داشته باشید شماره اندیس های آرایه از عدد صفر شروع می شود . بنابراین برای دسترسی به اولین آیتم آرایه باید اندیس آرایه را عدد 0 قرار دهید و برای دسترسی به آخرین آیتم آرایه باید از عدد -1 برای اندیس آرایه استفاده نمایید .

```
print $days[-1]; # outputs Sun  
print $days[-7]; # outputs Mon
```

ایجاد آرایه ای از اعداد مرتب

در زبان برنامه نویسی پرل برای ایجاد آرایه ای از آیتم های مرتب (به عنوان نمونه اعداد طبیعی مرتب یک تا ده) بجای تاپی یکی یکی آن ها برای راحتی بیشتر می توان از عملگر (..) استفاده نمود برای روشن شدن موضوع به مثال زیر توجه نمایید

```
#!/usr/bin/perl

@var_10 = (1..10);
@var_20 = (10..20);
@var_abc = (a..z);

print "@var_10\n";    # Prints number from 1 to 10
print "@var_20\n";    # Prints number from 10 to 20
print "@var_abc\n";   # Prints number from a to z
```

اجرای کد بالا نتیجه زیر را ظاهر خواهد نمود

```
1 2 3 4 5 6 7 8 9 10
10 11 12 13 14 15 16 17 18 19 20
a b c d e f g h i j k l m n o p q r s t u v w x y z
```

اندازه آرایه

تعداد آیتم هایی که در یک آرایه وجود دارند اندازه آرایه نامیده می شود . برای بدست آوردن اندازه یک آرایه باید نام یک آرایه را روبه روی تساوی و یک متغیر اسکالر قرار دهید که در این صورت اندازه آرایه در آن متغیر اسکالر ذخیره می گردد . نام یک آرایه به صورت `@array` می باشد حال برای به دست آوردن آخرین اندیس یک آرایه باید از علامت `$#` استفاده نمایید یعنی نام آرایه به این `$#array` صورت می گردد برای روشن شدن موضوع به مثال زیر توجه نمایید .

```
#!/usr/bin/perl

@array = (1,2,3);
$array[50] = 4;

$size = @array;
$max_index = $#array;

print "Size: $size\n";
print "Max Index: $max_index\n";
```

اجرای کد بالا نتیجه زیر را ظاهر خواهد نمود

```
Size: 51
Max Index: 50
```

حذف و اضافه آیتم ها در آرایه

زبان برنامه نویسی پرل توابع زیادی را برای عملیات مختلف بر روی آرایه ها فراهم می نماید . یکی از این عملیات ها بر وی آرایه اضافه کردن یک آیتم به آرایه و عملیات دیگر حذف یک آیتم از آرایه می باشد برای اضافه کردن یک آیتم به یک آرایه از تابع `push` و

unshift استفاده می شود و برای حذف یک آیتیم از آرایه از تابع pop و shift استفاده می گردد . تابع push یک آیتیم را به آخر آرایه اضافه می نماید و تابع unshift یک آیتیم را به ابتدای آرایه ضافه می نماید . تابع pop یک آیتیم را از آخر آرایه حذف می نماید و تابع shift یک عنصر را از ابتدای آرایه حذف می نماید . برای یادگیری کاربرد این توابع به مثال زیر توجه نمایید .

```
#!/usr/bin/perl

# create a simple array
@coins = ("Quarter","Dime","Nickel");
print "1. \@coins = \@coins\n";

# add one element at the end of the array
push(@coins, "Penny");
print "2. \@coins = \@coins\n";

# add one element at the beginning of the array
unshift(@coins, "Dollar");
print "3. \@coins = \@coins\n";

# remove one element from the last of the array.
pop(@coins);
print "4. \@coins = \@coins\n";

# remove one element from the beginning of the array.
shift(@coins);
print "5. \@coins = \@coins\n";
```

اجرای کد بالا نتیجه زیر را ظاهر خواهد نمود

```
1. @coins = Quarter Dime Nickel
2. @coins = Quarter Dime Nickel Penny
3. @coins = Dollar Quarter Dime Nickel Penny
4. @coins = Dollar Quarter Dime Nickel
5. @coins = Quarter Dime Nickel
```

استخراج آیتیم ها از آرایه

در زبان برنامه نویسی پرل شما می توانید یک یا هر چندتا آیتیم که دوست داشته باشید از آرایه ها استخراج نمایید برای روشن شدن موضوع به مثال زیر توجه نمایید

```
#!/usr/bin/perl

@days = qw/Mon Tue Wed Thu Fri Sat Sun/;

@weekdays = @days[3,4,5];

print "@weekdays\n";
```


اجرای کد بالا نتیجه زیر را ظاهر خواهد نمود

```
Thu Fri Sat
```

شما حتی می توانید یک دسته از آیتم ها را استخراج نمایید برای روشن شدن موضوع به مثال زیر توجه نمایید .

```
#!/usr/bin/perl

@days = qw/Mon Tue Wed Thu Fri Sat Sun/;

@weekdays = @days[3..5];

print "@weekdays\n";
```

اجرای کد بالا نتیجه زیر را ظاهر خواهد نمود

```
Thu Fri Sat
```

تغییر آیتم های آرایه

برای تغییر تعدادی از آیتم های یک آرایه می توانید از تابع **splice()** استفاده نمایید این تابع 4 تا آرگومان دارد که در زیر نام آن ها را می بینید

```
splice @ARRAY, OFFSET [ , LENGTH [ , LIST ] ]
```

این تابع آیتم های آرایه @ARRAY را از آیتم شماره OFFSET به تعداد LENGTH آیتم با آیتم ها آرایه LIST جایگزین می نماید برای روشن شدن موضوع به مثال زیر توجه نمایید

```
#!/usr/bin/perl

@nums = (1..20);
print "Before - @nums\n";

splice(@nums, 5, 5, 21..25);
print "After - @nums\n";
```

اجرای کد بالا نتیجه زیر را ظاهر خواهد نمود

```
Before - 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
After - 1 2 3 4 5 21 22 23 24 25 11 12 13 14 15 16 17 18 19 20
```

در این مثال 5 تا آیتم آرایه از آیتم شماره 6 در واقع آتم های 6 تا 10 با شماره های 21, 22, 23, 24 و 25 جایگزین شد .

تبدیل رشته به آرایه

در پرل تابعی وجود دارید به نام **split()** که می تواند بر اساس الگویی که برای آن تعیین می نمایید یک رشته را تجزیه نماید و در یک آرایه قرار دهد آرگومان های این تابع در زیر آمده است .

```
split [ PATTERN [ , EXPR [ , LIMIT ] ] ]
```

در بالا PATTERN الگوی تجزیه رشته و EXPR رشته مورد نظر می باشد . این تابع یک رشته را به صورت جدا جدا در یک آرایه قرار می دهد به عنوان نمونه برای قرار دادن کلمات یک جمله در یک آرایه به صورتی که هر کلمه در یک آیتم آرایه قرار بگیرد باید از این تابع به همراه الگوی خالی استفاده نمایید برای روشن شدن بیشتر موضوع به مثال زیر توجه نمایید .

```
#!/usr/bin/perl

# define Strings
$var_string = "Rain-Drops-On-Roses-And-Whiskers-On-Kittens";
$var_names = "Larry,David,Roger,Ken,Michael,Tom";

# transform above strings into arrays.
@string = split('-', $var_string);
@names = split(',', $var_names);

print "$string[3]\n"; # This will print Roses
print "$names[4]\n"; # This will print Michael
```

اجرای کد بالا نتیجه زیر را ظاهر خواهد نمود

```
Roses
Michael
```

تبدیل آرایه به رشته

در پرل برای تبدیل یک آرایه به رشته از تابع **join()** استفاده می شود آرگومان های این تابع در زیر آورده شده است

```
join EXPR, LIST
```

در این تابع قسمت `EXPR` رشته ای می باشد که تصمیم داریم آیتم های آرایه را با آن به هم متصل نماییم و به یک رشته تبدیل نماییم و قسمت `LIST` آرایه ای می باشد که قصد داریم آن را به یک رشته تبدیل نماییم . برای یادگیری نحوه کاربرد این تابع به مثال زیر توجه نمایید

```
#!/usr/bin/perl

# define Strings
$var_string = "Rain-Drops-On-Roses-And-Whiskers-On-Kittens";
$var_names = "Larry,David,Roger,Ken,Michael,Tom";

# transform above strings into arrays.
@string = split('-', $var_string);
@names = split(',', $var_names);

$string1 = join( '-', @string );
$string2 = join( ',', @names );

print "$string1\n";
print "$string2\n";
```

اجرای کد بالا نتیجه زیر را ظاهر خواهد نمود

```
Rain-Drops-On-Roses-And-Whiskers-On-Kittens
Larry,David,Roger,Ken,Michael,Tom
```

مرتب کردن آرایه ها

مرتب کردن یکی از مهمترین کاربرد کامپیوترها می باشد . در زبان برنامه نویسی پرل برای مرتب کردن از تابع `sort()` استفاده می گردد

```
sort [ SUBROUTINE ] LIST
```

این تابع یک تابع کم آرگومان می باشد . الگوی آرگومان های آن در بالا نشان داده شده است . این تابع یک آرایه را به عنوان آرگومان می گیرد و یک آرایه مرتب بر می گرداند . برای روشن شدن موضوع به مثال زیر توجه نمایید

```
#!/usr/bin/perl

# define an array
@foods = qw(pizza steak chicken burgers);
print "Before: @foods\n";

# sort this array
@foods = sort(@foods);
print "After: @foods\n";
```

اجرای کد بالا نتیجه زیر را ظاهر خواهد نمود

```
Before: pizza steak chicken burgers
After:  burgers chicken pizza steak
```

شاید از خودتان بپرسید که در مثال بالا آرایه شامل اعداد نیست پس چطور تابع **sort()** مرتب کرده است ولی این تابع ابتدا حرف اول هر کلمه را به کد اسکی تبدیل می نماید سپس بر اساس عدد آن را مرتب می نماید .

تغییر شماره اندیس ابتدایی آرایه

می دانیم که در زبان برنامه نویسی پرل اندیس آیتم های آرایه ها از صفر شروع می شود . ولی در برنامه نویسی بعضی اوقات نیاز می شود که اندیس آرایه از صفر شروع نشود به عنوان مثال از یک یا دو شروع بشود . برای این کار از علامت `$[` استفاده می گردد برای یادگیری کاربرد این علامت `$[` در زبان برنامه نویسی پرل به مثال زیر توجه نمایید

```
#!/usr/bin/perl

# define an array
@foods = qw(pizza steak chicken burgers);
print "Foods: @foods\n";

# Let's reset first index of all the arrays.
$[ = 1;

print "Food at \@foods[1]: $foods[1]\n";
print "Food at \@foods[2]: $foods[2]\n";
```

اجرای کد بالا نتیجه زیر را ظاهر خواهد نمود

```
Foods: pizza steak chicken burgers
Food at \@foods[1]: pizza
Food at \@foods[2]: steak
```

ادغام آرایه ها

آیتم های یک آرایه که با کاما از هم جدا می شوند می شود براحتی می توان یکجا تمام آیتم های آن ها را نمایش داد به عبارتی آن ها را در یک مقدار ادغام کرد برای روشن شدن موضوع به مثال زر توجه نمایید

```
#!/usr/bin/perl

@numbers = (1,3,(4,5,6));

print "numbers = @numbers\n";
```

اجرای کد بالا نتیجه زیر را ظاهر خواهد نمود

```
numbers = 1 3 4 5 6
```

بیاد داشته باشید هر کدام از آیتم های یک آرایه خود می توانند یک آرایه باشند برای روشن شدن موضوع به مثال زیر توجه نمایید

```
#!/usr/bin/perl

@odd = (1,3,5);
@even = (2, 4, 6);

@numbers = (@odd, @even);

print "numbers = @numbers\n";
```

اجرای کد بالا نتیجه زیر را ظاهر خواهد نمود

```
numbers = 1 3 5 2 4 6
```

انتخاب آیتم ها از لیست ها

برای استخراج یک آیتم از یک آرایه می توانید از علامت براکت در جلوی آرایه به همراه شماره اندیس آیتم مورد نظر استفاده کرد برای روشن شدن موضوع به مثال زیر توجه نمایید

```
#!/usr/bin/perl

$var = (5,4,3,2,1)[4];

print "value of var = $var\n"
```

اجرای کد بالا نتیجه زیر را ظاهر خواهد نمود

```
value of var = 1
```

به صورت مشابه می توان تعدادی از آیتم ها را از آرایه جدا نمود برای روشن شدن موضوع به مثال زیر توجه نمایید

```
#!/usr/bin/perl
@list = (5,4,3,2,1)[1..3];
print "Value of list = @list\n";
```

اجرای کد بالا نتیجه زیر را ظاهر خواهد نمود

```
Value of list = 4 3 2
```

فادی کیپامدتی