

acetinkaya /
VeriYapilari-ders4

<> Code

Issues

Pull requests

Actions

Projects

Security

Insights

Settings



MIT license

4 stars

0 forks

1 watching

Branches

Activity

Tags

Public repository



1 Branch

0 Tags



Go to file



Go to file



Add file

<> Code



acetinkaya Update README.md

de6316f · 2 months ago



Asci.png

Add files via upload

2 months ago



LICENSE

Initial commit

3 months ago



Operator_tablosu-1.png

Add files via upload

3 months ago



Operator_tablosu-2.png

Add files via upload

3 months ago



Operator_tablosu-3.png

Add files via upload

3 months ago



Operator_tablosu-4.png

Add files via upload

3 months ago



Programlama-2.png

Add files via upload

3 months ago



Programlama-3.png

Add files via upload

2 months ago



Programlama-4.png

Add files via upload

2 months ago



README.md

Update README.md

2 months ago



README



MIT license



Python - Veri Yapıları ve Algoritmalar

Ders 4. Programlama Dilinde Veri Türleri ve Operatörler



Bu ders içerisinde;

- 4.1. Veri Türleri
 - 4.1.1. Integer (int) Veri Türü
 - 4.1.2. Float (float) Veri Türü
 - 4.1.3. String (str) Veri Türü
 - 4.1.4. Boolean (bool) Veri Türü
- 4.2. Operatör Nedir?
 - 4.2.1. Aritmetik Operatörler
 - 4.2.2. Karşılaştırma Operatörleri
 - 4.2.3. Mantıksal Operatörler
 - 4.2.4. Atama, Arttırma ve Azaltma Operatörleri
 - 4.2.5. Bit Düzeyinde Operatörler
- 4.3. Operatör İşlem Önceliği
- 4.4. ASCII Tablosu
- 4.5. Uygulamalı Örnekler ve Alıştırmalar

4.1. Veri Türleri

- 4.1.1. Integer (int) Veri Türü -> Tam sayıları temsil eder. Matematiksel işlemler ve sayısal veriler için kullanılır.
- 4.1.2. Float (float) Veri Türü -> Ondalıklı (kesirli) sayıları ifade eder.
- 4.1.3. String (str) Veri Türü -> Metin verilerini saklar. Harfler, kelimeler, cümleler veya özel karakterler içerebilir.
- 4.1.4. Boolean (bool) Veri Türü -> Koşulları kontrol etmek için kullanılır. Sadece iki değer alabilir: "True" (Doğru - 1) veya "False" (Yanlış - 0).

4.2. Operatör Nedir?

Operatörler, programlama dillerinde veri üzerinde işlem yapmak için kullanılan sembollerdir. Python'da operatörler, değişkenler ve değerler üzerinde işlemler gerçekleştirmek için kullanılmaktadır.

4.2.1. Aritmetik Operatörler

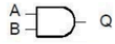
Operatör	İşlevi	Örnek Kullanımı	Çıktısı
+	Toplama	$7 + 3$	10
-	Çıkarma	$7 - 3$	4
*	Çarpma	$7 * 3$	21
/	Bölme	$7 / 3$	2,333
%	Mod - Kalan Bulma	$7 \% 3$	1
**	Üs Alma	$7 ** 3$	343
Öğr. Gör. Ali ÇETİNKAYA - 2024			

4.2.2. Karşılaştırma Operatörleri

İlgili işlemdeki değerleri karşılaştırarak boolean "True" veya "False" sonuç elde edilmektedir.

Operatör	İşlevi	Örnek Kullanımı	Çıktısı
==	Eşit mi?	$7 == 3$	False
!=	Eşit değil mi?	$7 != 3$	True
>	Büyük mü?	$7 > 3$	True
<	Küçük mü?	$7 < 3$	False
>=	Büyük veya eşit mi?	$7 >= 3$	True
<=	Küçük veya Eşit mi?	$7 <= 3$	False
Öğr. Gör. Ali ÇETİNKAYA - 2024			

4.2.3. Mantıksal Operatörler

**VE
İŞLEMİ
(AND)**

A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

**VEYA
İŞLEMİ
(OR)**

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

**ÖZEL VEYA
İŞLEMİ
(XOR)**

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

**DEĞİL
İŞLEMİ
(NOT)**

A	Q
0	1
1	0

Öğr. Gör. Ali ÇETİNKAYA - Eylül 2024

4.2.4. Atama, Arttırma ve Azaltma Operatörleri

Arttırma ve Azaltma Operatörleri

sayi_degiskeni = 5

toplam = sayi_degiskeni + 5

toplam += 5 ---> toplam = toplam + 5 ---> toplam değişkenine eşitliğin sağındaki 5 değerini ekle ve sonucu yine toplam değişkenine ata.

toplam -= 5 ---> toplam = toplam - 5 ---> toplam değişkenine eşitliğin sağındaki 5 değerini çıkar ve sonucu yine toplam değişkenine ata.

toplam *= 5 ---> toplam = toplam * 5 ---> toplam değişkenine eşitliğin sağındaki 5 değerini çarp ve sonucu yine toplam değişkenine ata.

toplam /= 5 ---> toplam = toplam / 5 ---> toplam değişkenine eşitliğin sağındaki 5 değerini böl ve sonucu yine toplam değişkenine ata.

toplam %= 5 ---> toplam = toplam % 5 ---> toplam değişkenine eşitliğin sağındaki 5 değerine bölümünden kalanı al ve sonucu yine toplam değişkenine ata.

Öğr. Gör. Ali ÇETİNKAYA - Eylül 2024

4.2.5. Bit Düzeyinde Operatörler

Bit düzeyinde (bitwise) operatörler, sayıların ikili (binary) temsilleri üzerinde işlem yapmak için kullanılır. Bu operatörler, sayıları bit seviyesinde işler ve genellikle düşük seviyeli programlama veya performans gerektiren uygulamalarda tercih edilmektedir.

Operatör	İşlevi	Örnek Kullanımı	Çıktısı
&	Mantıksal Ve	7 & 3	3
	Mantıksal Veya	7 3	7
^	Özel VEYA (XOR)	7 ^ 3	4
~	Değil	7 ~ 3	-4
<<	Sola Kaydırma	7 << 3	56
>>	Sağa Kaydırma	7 >> 3	0
Öğr. Gör. Ali ÇETİNKAYA - 2024			

4.3. Operatör İşlem Önceliği

İşlem Önceliği	Operatör
1	a++, a--, [], ()
2	++a, --a
3	*, /, %
4	+, -
5	<<, >>
6	<, >, <=, >=
7	==, !=
8	&
9	^
10	
11	&&
12	
13	=

Öğr. Gör. Ali ÇETİNKAYA - 2024

4.4. ASCII Tablosu

✿	0	1	2	3	4	5	6	7	8	9
0	nul	soh	stx	etx	eot	enq	ack	bel	bs	ht
1	nl	vt	ff	cr	so	si	dle	dc1	dc2	dc3
2	dc4	nak	syn	etb	can	em	sub	esc	fs	gs
3	rs	us	<u>b</u>	!	"	#	\$	%	&	'
4	()	*	+	,	-	.	/	0	1
5	2	3	4	5	6	7	8	9	:	;
6	<	=	>	?	@	A	B	C	D	E
7	F	G	H	I	J	K	L	M	N	O
8	P	Q	R	S	T	U	V	W	X	Y
9	Z	[\]	^	_	'	a	b	c
10	d	e	f	g	h	i	j	k	l	m
11	n	o	p	q	r	s	t	u	v	w
12	x	y	z	{		}	~	del		

Öğr. Gör. Ali ÇETİNKAYA - 2024

4.5. Uygulamalı Örnekler ve Alıştırmalar

Örnek - 1:

```
Ders_no = 4    # Integer (int) Veri Türü
pi_sayisi = 3.14159 # Float (float) Veri Türü
Dersin_ismi = "Python - Veri Yapıları ve Algoritmalar" # String (str) Veri Türü
durum1 = True
durum2 = False

print(Ders_no)
print(pi_sayisi)
print(Dersin_ismi)
print(durum1)
print(durum2)
```

Örnek - 2:

```
a = 7
b = 3
print(a + b)
```

Örnek - 3:

```
a = 7
b = 3
print("Toplama:", a * b)
print("Üs alma:", a ** b)
print("Mod:", a % b)
```

Örnek - 4:

```
a = 7
b = 3
print("a == b:", a == b)
print("a < b:", a < b)
```

Örnek - 5:

```
a = True
b = False
print("a and b:", a and b)
print("a or b:", a or b)
print("not a:", not a)
```

Örnek - 6:

```
x = 7
x += 3 # x = x + 3
print(x)
```



Örnek - 7:

```
a = 7
b = 3
print("a & b:", a & b)
print("a << b:", a << b)
```



Örnek - 8:

```
print(7 & 3) # 3
print(7 | 3) # 7
print(7 ^ 3) # 4
print(~3)    # -4
print(7 << 3) # 56
print(7 >> 3) # 2
```



Releases

No releases published

[Create a new release](#)

Packages

No packages published

[Publish your first package](#)