FAKULTET INŽENJERSKIH NAUKA
UNIVERZITET U KRAGUJEVCU

Tema: Programiranje aplikacija IOS sitema
Student: Nevena Stašić
Profesor: Nenad Grujović

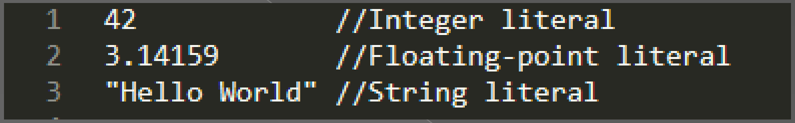
Kragujevac, 2019. godine

Sadržaj

- Literals
- Operatori
- Aritmetički operatori
- Relacioni operatori
- Logički operatori
- Binarni operatori
- Operatori dodeljivanja
- Operatori ranga

Literals

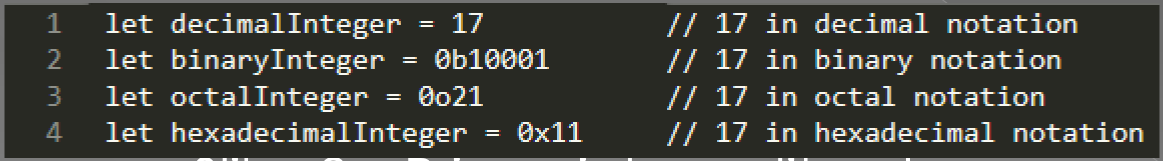
Literal je izvorni kod koji predstavlja vrednost integera, vrednost floating-point broja ili vrednost string tipa podataka.



```
42            //Integer literal
3.14159       //Floating-point literal
"Hello World" //String literal
```

Slika 1 – Primeri literala

Literali integer tipa podatka, mogu biti decimalne, binarne, oktalne ili heksadecimalne konstante.



```
let decimalInteger = 17           // 17 in decimal notation
let binaryInteger = 0b10001       // 17 in binary notation
let octalInteger = 0o21           // 17 in octal notation
let hexadecimalInteger = 0x11     // 17 in hexadecimal notation
```

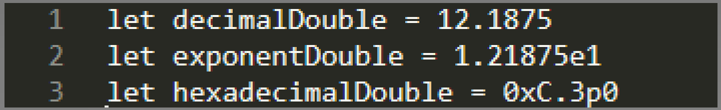
Slika 2 – Primer integer literala

Literals

Floating – point literali imaju celobrojni deo, decimalnu tačku, frakcioni deo i ekponencijalni deo. Floatin – point literale možemo predstaviti sa pomeračkim zarezom ili u decimalnom obliku ili u heksadecimalnom obliku.

Decimalni floating – point literali se sastoje od niza decimalnih cifri, pračenih ili decimalnom frakcijom ili decimalnim eksponentom ili i jednim i drugim.

Heksadecimalni floating-point literali se sastoje od prefiksa 0x, iza koga sledi opcionalna heksadecimalna frakcija, praćena heksadecimalnim eksponentom.



```
let decimalDouble = 12.1875
let exponentDouble = 1.21875e1
let hexadecimalDouble = 0xC.3p0
```

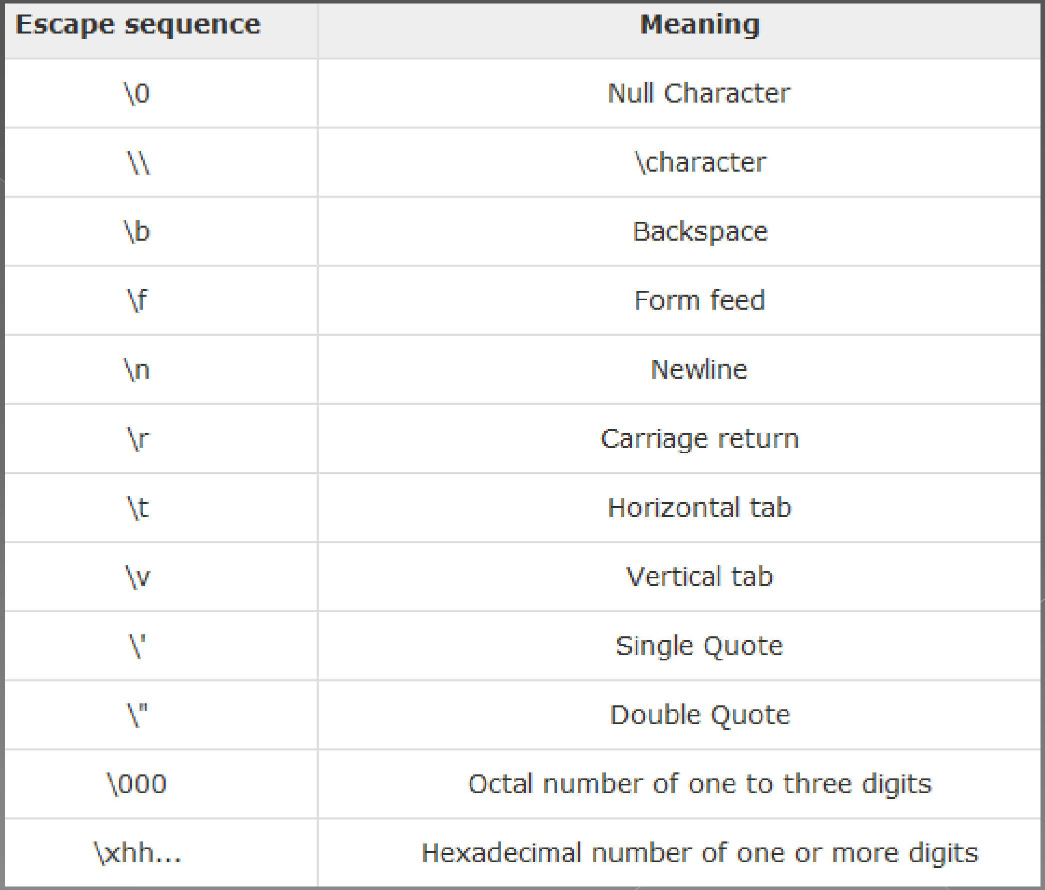
Slika 3 – Primeri floatin-point literala

Literals

String literal je niz karaktera koji su okruženi dvostrukim navodnicima. (Primer: “character”)

String literali ne mogu da sadrže nezaštićeni dvostruki citat (“) i nezaštićenu obrnutu kosu crtu (\).

Specijalni znakovi mogu biti uključeni u string literale koristeći sledeće sekvence:

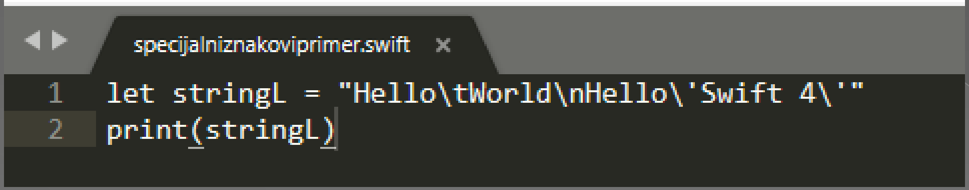


Escape sequence	Meaning
\0	Null Character
\\	\character
\b	Backspace
\f	Form feed
\n	Newline
\r	Carriage return
\t	Horizontal tab
\v	Vertical tab
\'	Single Quote
\"	Double Quote
\000	Octal number of one to three digits
\xhh...	Hexadecimal number of one or more digits

Slika 4 – specijalni znakovi

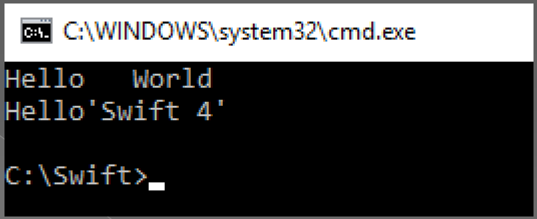
Primer

- Primer korišćenja specijalnih znakova sa prethodnog slajda, prilikom štampanja nekih vrednosti:



```
let stringL = "Hello\tWorld\nHello\'Swift 4\'"
print(stringL)
```

Slika 5 - Kod



```
Hello   World
Hello'Swift 4'

C:\Swift>
```

Slika 6 – Startovan program

Literals

Od Boolean literala, postoji tri vrste literala i oni su deo standardnih Swift 4 ključnih reči:

- True – prezentuje kao i u svim jezicima true vrednost
- False – takođe kao i u svim jezicima prezentuje false vrednost
- Nil – ovo je null vrednost

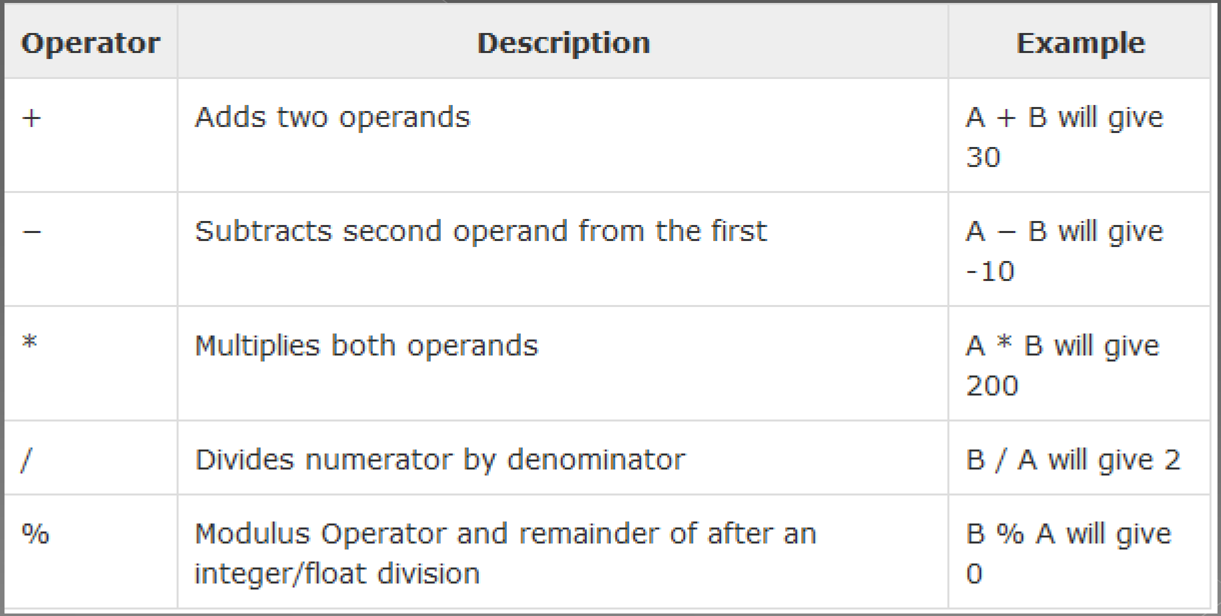
Operatori

Operator je simbol koji saopštava kompajleru da izvodi određene matematičke ili logičke obrasce. Objective C je bogat ugrađenim operatorima i pruža sledeće tipove operatora:

- Aritmerički operatori;
- Relacioni operatori;
- Logički operatori;
- Binarni operatori;
- Operatori dodeljivanja;
- Operatori ranga;

Aritmetički operatori

- Sledeća tabela prikazuje sve aritmetičke operatore podržane Swift 4 programskim jezikom. Pretpostavimo da promenljiva A ima vrednost 10, dok promenljiva B ima vrednost 20, onda je:

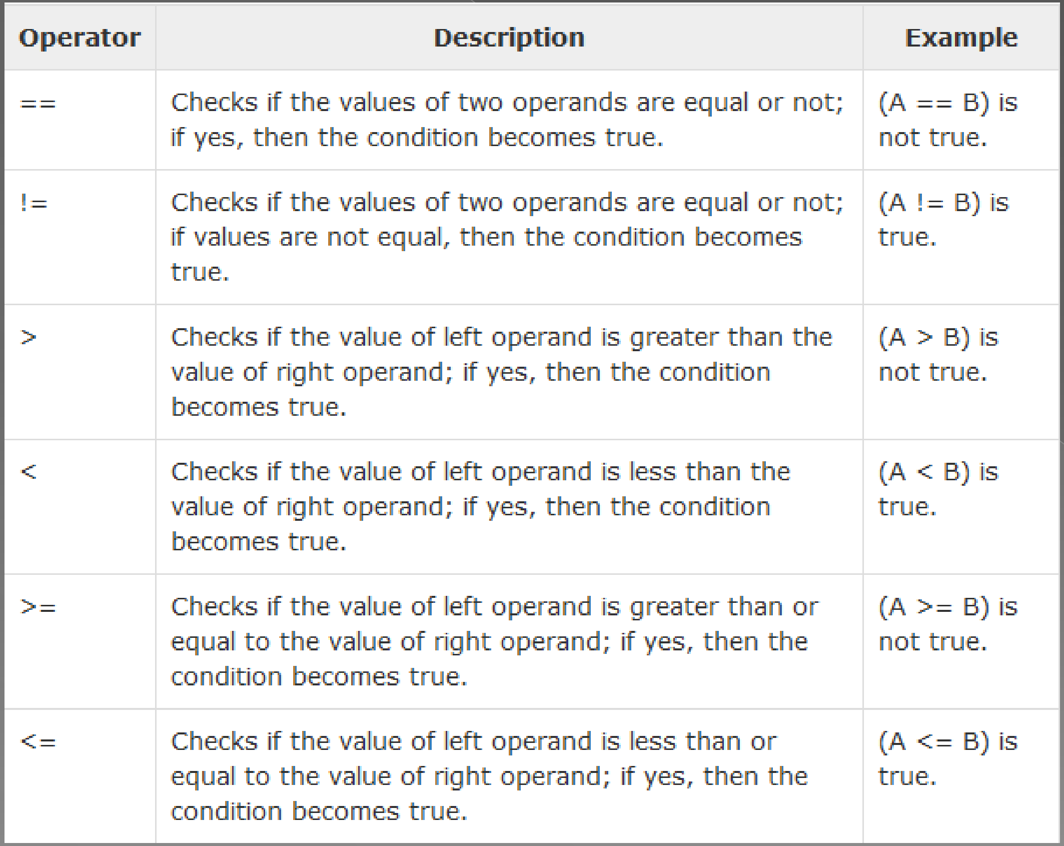


Operator	Description	Example
+	Adds two operands	A + B will give 30
−	Subtracts second operand from the first	A − B will give -10
*	Multiplies both operands	A * B will give 200
/	Divides numerator by denominator	B / A will give 2
%	Modulus Operator and remainder of after an integer/float division	B % A will give 0

Slika 7 – Aritmetički operatori

Relacioni operatori

- Sledeća tebela prikazuje sve operatore poređenja (relacija), koje podržava Swift 4 programski jezik. Kao i u prethodnoj tabeli, pretpostavićemo da je promenljiva A jednaka 10 , a promenljiva B jednaka 20, tada:

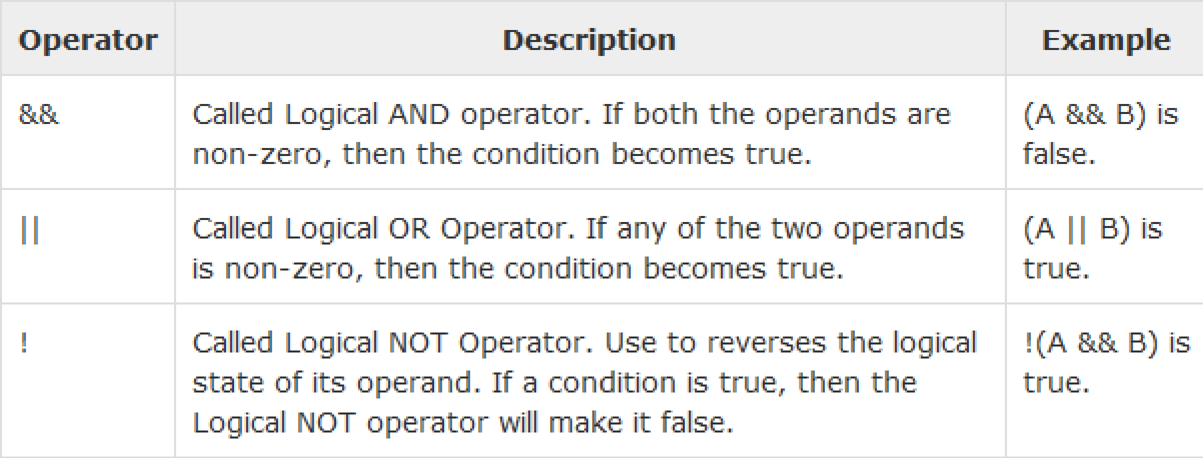


Operator	Description	Example
==	Checks if the values of two operands are equal or not; if yes, then the condition becomes true.	(A == B) is not true.
!=	Checks if the values of two operands are equal or not; if values are not equal, then the condition becomes true.	(A != B) is true.
>	Checks if the value of left operand is greater than the value of right operand; if yes, then the condition becomes true.	(A > B) is not true.
<	Checks if the value of left operand is less than the value of right operand; if yes, then the condition becomes true.	(A < B) is true.
>=	Checks if the value of left operand is greater than or equal to the value of right operand; if yes, then the condition becomes true.	(A >= B) is not true.
<=	Checks if the value of left operand is less than or equal to the value of right operand; if yes, then the condition becomes true.	(A <= B) is true.

Slika 8 – Relacioni operatori

Logički operatori

- Tabela ispod, nam prikazuje sve logičke operatore podržane u Swift 4 programskom jeziku. Pretpostavimo da promenljiva A ima vrednost 1, dok promenljiva B ima vrednost 0. Tada:

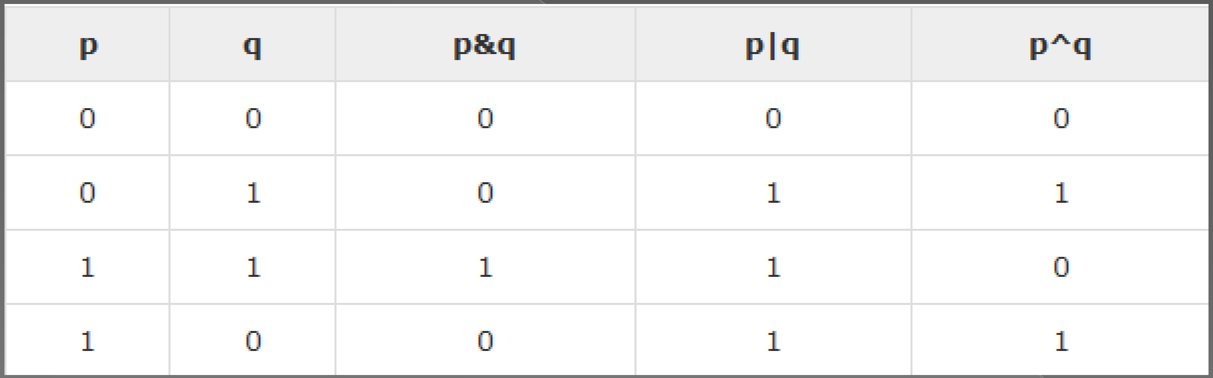


Operator	Description	Example
&&	Called Logical AND operator. If both the operands are non-zero, then the condition becomes true.	(A && B) is false.
||	Called Logical OR Operator. If any of the two operands is non-zero, then the condition becomes true.	(A || B) is true.
!	Called Logical NOT Operator. Use to reverses the logical state of its operand. If a condition is true, then the Logical NOT operator will make it false.	!(A && B) is true.

Slika 9 – Logički operatori

Binarni operatori

- Ovi operatori izvode operacije bit po bit. Tabele istine za operacije &, | i ^ su sledeće:

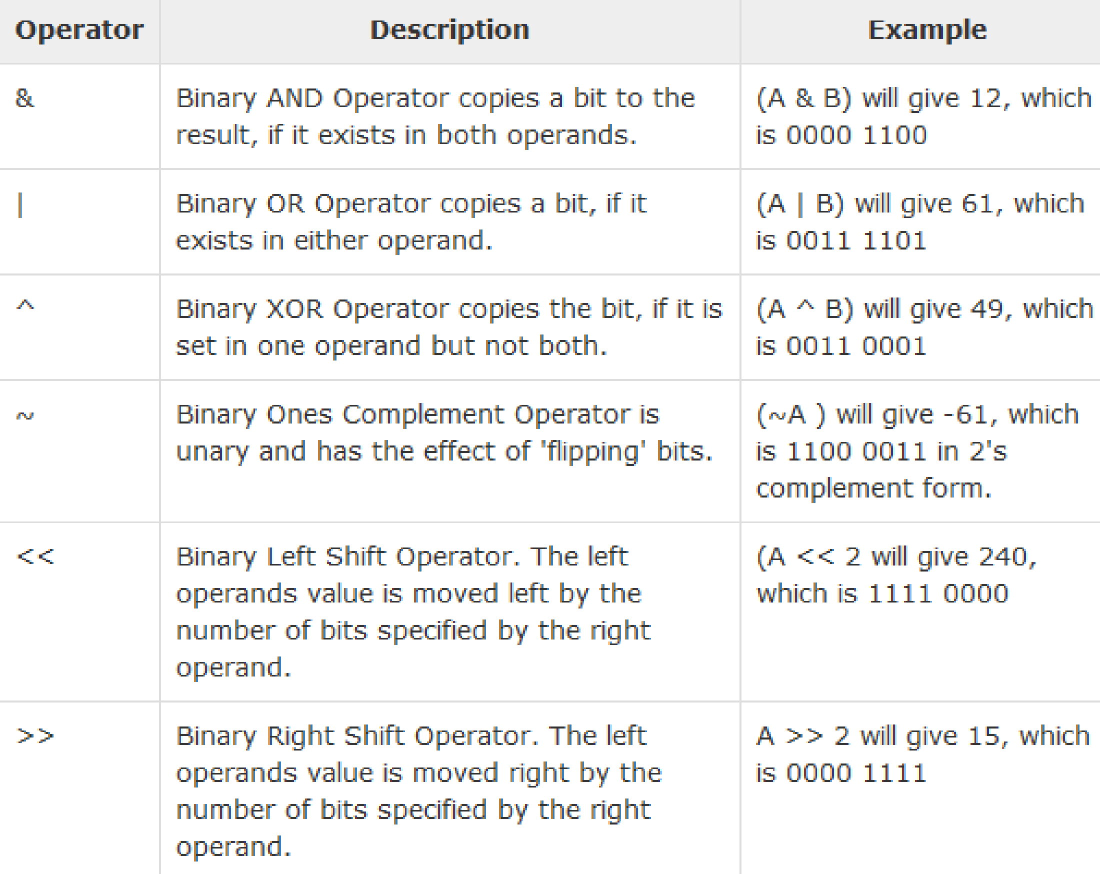


p	q	p&q	p|q	p^q
0	0	0	0	0
0	1	0	1	1
1	1	1	1	0
1	0	0	1	1

Slika 10 – Binarni operatori

Binarni operatori

- Binarni operatori podržani u Swift 4 programskom jeziku su navedeni u tabeli ispod. Pretpostavimo da promenljiva A ima vrednost 60 i da promenljiva B ima vrednost 12, a zatim 7:

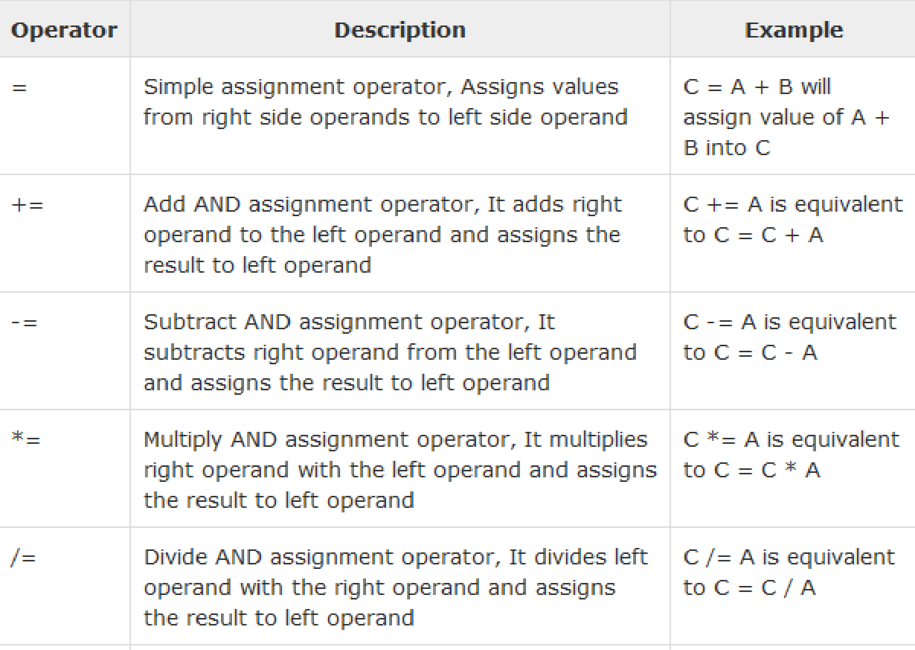


Operator	Description	Example
&	Binary AND Operator copies a bit to the result, if it exists in both operands.	(A & B) will give 12, which is 0000 1100
|	Binary OR Operator copies a bit, if it exists in either operand.	(A | B) will give 61, which is 0011 1101
^	Binary XOR Operator copies the bit, if it is set in one operand but not both.	(A ^ B) will give 49, which is 0011 0001
~	Binary Ones Complement Operator is unary and has the effect of 'flipping' bits.	(~A) will give -61, which is 1100 0011 in 2's complement form.
<<	Binary Left Shift Operator. The left operands value is moved left by the number of bits specified by the right operand.	(A << 2 will give 240, which is 1111 0000
>>	Binary Right Shift Operator. The left operands value is moved right by the number of bits specified by the right operand.	A >> 2 will give 15, which is 0000 1111

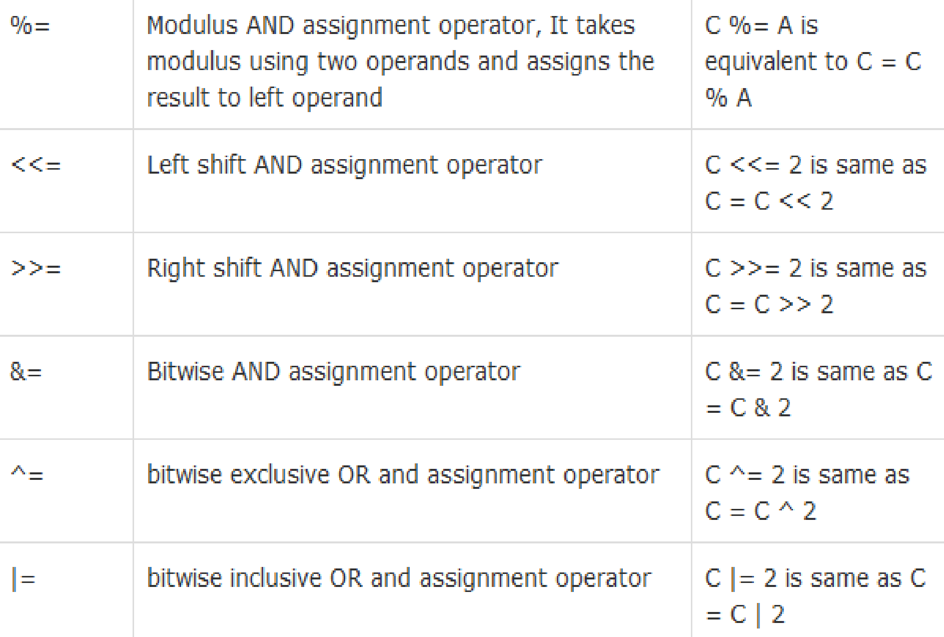
Slika 11 – Binarni operatori u Swiftu 4

Operatori dodeljivanja

- Swift 4 podržava sledeće operatore dodeljivanja:



Operator	Description	Example
=	Simple assignment operator, Assigns values from right side operands to left side operand	C = A + B will assign value of A + B into C
+=	Add AND assignment operator, It adds right operand to the left operand and assigns the result to left operand	C += A is equivalent to C = C + A
-=	Subtract AND assignment operator, It subtracts right operand from the left operand and assigns the result to left operand	C -= A is equivalent to C = C - A
*=	Multiply AND assignment operator, It multiplies right operand with the left operand and assigns the result to left operand	C *= A is equivalent to C = C * A
/=	Divide AND assignment operator, It divides left operand with the right operand and assigns the result to left operand	C /= A is equivalent to C = C / A

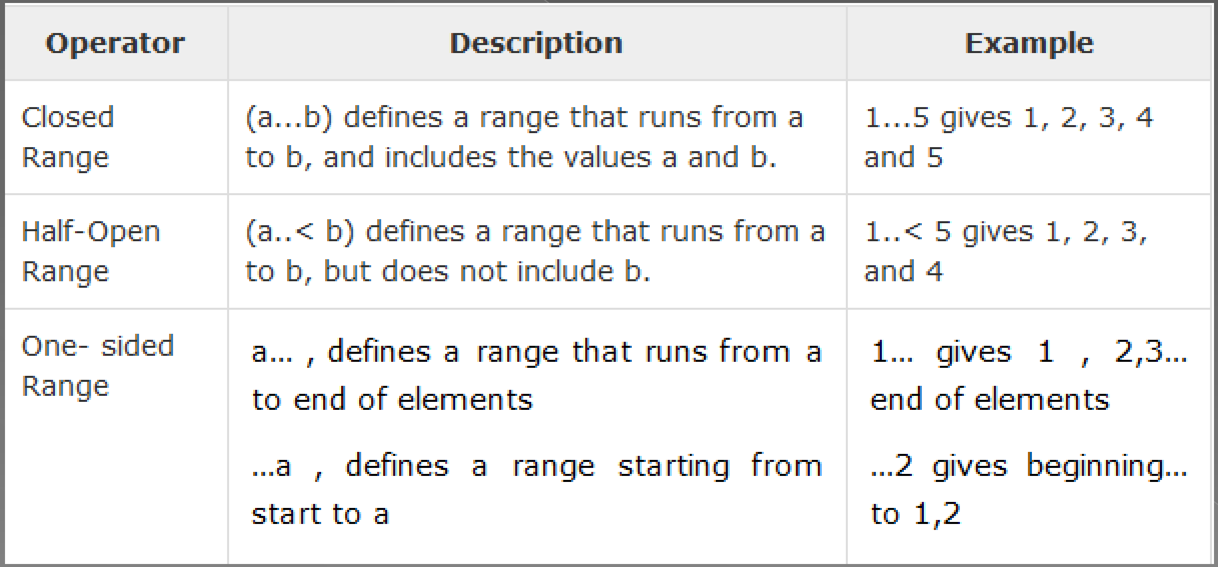


%=	Modulus AND assignment operator, It takes modulus using two operands and assigns the result to left operand	C %= A is equivalent to C = C % A
<<=	Left shift AND assignment operator	C <<= 2 is same as C = C << 2
>>=	Right shift AND assignment operator	C >>= 2 is same as C = C >> 2
&=	Bitwise AND assignment operator	C &= 2 is same as C = C & 2
^=	bitwise exclusive OR and assignment operator	C ^= 2 is same as C = C ^ 2
|=	bitwise inclusive OR and assignment operator	C |= 2 is same as C = C | 2

Slike 12 i 13 – Operatori dodeljivanja

Operatori ranga

- Swift 4 programski jezik, uključuje dva operatora raspona, kosi su prečice za izražavanje raspona vrednosti. Sledeća tebela objašnjava ova dva operatora:



Operator	Description	Example
Closed Range	(a...b) defines a range that runs from a to b, and includes the values a and b.	1...5 gives 1, 2, 3, 4 and 5
Half-Open Range	(a..< b) defines a range that runs from a to b, but does not include b.	1..< 5 gives 1, 2, 3, and 4
One- sided Range	a… , defines a range that runs from a to end of elements …a , defines a range starting from start to a	1… gives 1 , 2,3… end of elements …2 gives beginning… to 1,2

Slika 14 – Operatori ranga