

Octal integer literal

An integer value with base 8 is called octal integer. This octal integer is created using all the digits from 0-7. This integer is prefix with 0o or 0O.

Applications of octal integer literals

1. Operating Systems
2. Assembly Language

```
>>> a=0o256
>>> a
174
>>> b=0o78
SyntaxError: invalid digit '8' in octal literal
>>> c=0O99
SyntaxError: invalid digit '9' in octal literal
>>> d=0o777
>>> d
511
>>> type(d)
<class 'int'>
```

Decimal to Octal	Octal to Decimal												
$(68)_{10} \longrightarrow (0o104)_8$ <table><tr><td>8</td><td>68</td><td></td></tr><tr><td>8</td><td>8</td><td>4</td></tr><tr><td>8</td><td>1</td><td>0</td></tr><tr><td></td><td></td><td>1</td></tr></table>	8	68		8	8	4	8	1	0			1	$(0o104)_8 \longrightarrow (68)_{10}$ $8^0 \times 4 + 8^1 \times 0 + 8^2 \times 1$ $4+0+64$
8	68												
8	8	4											
8	1	0											
		1											

oct() : oct() is a base conversion function, which is used to convert other types of integers to octal.

Decimal \rightarrow Octal

Hexadecimal → Octal
Binary → Octal

Example:

```
>>> oct(68)
'0o104'
```

Hexadecimal Integer Literal

An integer value with base 16 is called hexadecimal integer. This integer consists of digits from 0-9 and a-f/A-F. This integer is prefix with 0x or 0X.

Applications of Hexadecimal Integer

1. Larger values are represented in hexadecimal format
2. Color Value
3. Memory Address
4. Unicode Values

```
>>> n1=0xa
>>> n1
10
>>> n2=0xb
>>> n2
11
>>> n3=0x5
>>> n3
5
>>> n4=0xab
>>> n4
171
>>> n5=0xff
>>> n5
255
>>> n6=0xgg
SyntaxError: invalid hexadecimal literal
>>> n7=0x1ab2
```

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
a b c d e f

Decimal to Hexadecimal	Hexadecimal to Decimal
$\begin{array}{c} (255)_{10} \xrightarrow{\hspace{1cm}} (0xff)_{16} \\ \\ \begin{array}{ c c c } \hline 16 & 255 & \\ \hline 16 & 15 & 15 \\ \hline & & 15 \\ \hline \end{array} \end{array}$	$\begin{array}{c} 10 \\ (0xff)_{16} \xrightarrow{\hspace{1cm}} (255)_{10} \\ \\ 16^0 \times f + 16^1 \times f \\ 1 \times 15 + 16 \times 15 \\ 15 + 240 \end{array}$

hex() : It is a base conversion function, this convert other integers into hexadecimal integer

decimal → hexadecimal
octal → hexadecimal
binary → hexadecimal

```
>>> a=10
>>> a
10
>>> oct(a)
'0o12'
>>> hex(a)
'0xa'
>>> b=255
>>> b
255
>>> oct(b)
'0o377'
>>> hex(b)
```

Binary Integer

An integer value with base 2 is called binary integer. This integer is created using two digits 0 and 1. This integer is prefix with 0b or 0B.

Applications of binary integer

1. Embedded Applications Logic Gates

```
>>> b1=0b1
>>> b1
1
>>> b2=0b0
>>> b2
0
>>> b3=0b10
>>> b3
2
>>> b4=0b101
>>> b4
5
>>> b5=0b1010
>>> b5
10
>>> b7=0b102
SyntaxError: invalid digit '2' in binary literal
```

bin(): base conversion function,
decimal → binary
hexadecimal → binary
octal → binary

Decimal to binary	Binary to Decimal																		
<p>(25) $\xrightarrow{10} 0b11001 \xrightarrow{2}$</p> <table><tr><td>2</td><td>25</td><td></td></tr><tr><td>2</td><td>12</td><td>1</td></tr><tr><td>2</td><td>6</td><td>0</td></tr><tr><td>2</td><td>3</td><td>0</td></tr><tr><td>2</td><td>1</td><td>1</td></tr><tr><td></td><td></td><td>1</td></tr></table>	2	25		2	12	1	2	6	0	2	3	0	2	1	1			1	<p>$\begin{array}{ccccccc} & 4 & 3 & 2 & 1 & 0 \\ (0b11001) & \xrightarrow{2} & (25) & \xrightarrow{10} & & \end{array}$</p> <p>$2^0 \times 1 + 2^1 \times 0 + 2^2 \times 0 + 2^3 \times 1 + 2^4 \times 1$</p> <p>$1 + 0 + 0 + 8 + 16$</p>
2	25																		
2	12	1																	
2	6	0																	
2	3	0																	
2	1	1																	
		1																	

Example:

```

>>> a=15
>>> bin(a)
'0b1111'
>>> b=0o12
>>> bin(b)
'0b1010'
>>> c=0xa
>>> bin(c)
'0b1010'
>>> d=0b1010
>>> hex(d)
'0xa'

```

float data type

float data type is used to reserve memory for float literal or value.

A float value is numeric value with fractional part or decimal part.
In python float values/literals are represented in two formats

1. Fixed notations
2. Scientific notation