

The **square root** and **raise a number to the power** in Python: three options

A **square root** of a number is some value that, when multiplied by itself, returns that same number.

The symbol that we use for square root, $\sqrt{}$, always means a positive square root however. So $\sqrt{16}$ is 4 (and not -4).

Python has **three ways** to get the positive square root of a number:

- The `math.sqrt()` function **returns the square root as a precise floating-point value**. (This function is what you need 95% of the time.)

For example: (результат витягу корня квадратного - дійсне число)

```
from math import sqrt
```

```
x = sqrt(12)
```

```
print(x)
```

```
# Returns: 3,464101615137755
```

- The `math.isqrt()` function **returns the square root as an integer value** (meaning, rounded down to a whole number).

For example: (результат витягу корня квадратного - ціле число)

```
from math import sqrt
```

```
x = isqrt(12)
```

```
print(x)
```

```
# Returns: 3 (and not 3,464101615137755)
```

- And, when we **raise a number** to the power of 0.5 we also get the square root. We can use Python's **exponent ($**$) operator** or **`pow()` function** for this.

For example: (підвести число в степінь 0.5 . Степінь $0,5$ це корінь квадратний)

```
x = pow(12, 0.5)
```

```
print(x)
```

```
# Returns: 3,464101615137755
```