**Python 3.x built-in functions**

| **Function** | **Description** | **Example** |
| --- | --- | --- |
| abs() | Absolute value | abs(-5) → 5 |
| all() | True if all elements are true | all([True, 1, 3]) → True |
| any() | True if any element is true | any([0, False, 2]) → True |
| bin() | Convert to binary | bin(10) → '0b1010' |
| bool() | Convert to boolean | bool(0) → False |
| bytearray() | Mutable bytes | bytearray(b'abc') |
| bytes() | Immutable bytes | bytes([65,66]) → b'AB' |
| callable() | Check if object is callable | callable(print) → True |
| chr() | Unicode → char | chr(65) → 'A' |
| classmethod() | Create class method | class A: @classmethod def f(cls): pass |
| compile() | Compile code | compile('x+1','<string>','eval') |
| complex() | Create complex number | complex(2,3) → (2+3j) |
| delattr() | Delete attribute | delattr(obj,'attr') |
| dict() | Create dictionary | dict(a=1,b=2) → {'a':1,'b':2} |
| dir() | List attributes | dir([]) → ['append', 'clear', ...] |
| divmod() | Quotient & remainder | divmod(10,3) → (3,1) |
| enumerate() | Index iterable | list(enumerate(['a','b'])) → [(0,'a'),(1,'b')] |
| eval() | Evaluate string | eval('2+3') → 5 |
| exec() | Execute code | exec("x=5") |
| filter() | Filter iterable | list(filter(lambda x:x>2,[1,2,3])) → [3] |
| float() | Convert to float | float('3.5') → 3.5 |
| format() | Format string | format(3.14159,'.2f') → '3.14' |
| frozenset() | Immutable set | frozenset([1,2,3]) |
| getattr() | Get attribute | getattr(obj,'attr') |
| globals() | Global variables dict | globals() |
| hasattr() | Check attribute | hasattr(obj,'attr') → True/False |
| hash() | Hash value | hash('abc') |
| help() | Built-in help | help(len) |
| hex() | Convert to hex | hex(255) → '0xff' |
| id() | Unique object id | id(5) |
| input() | User input | input("Enter: ") |
| int() | Convert to int | int('10') → 10 |
| isinstance() | Check instance | isinstance(5,int) → True |
| issubclass() | Check subclass | issubclass(bool,int) → True |
| iter() | Create iterator | iter([1,2]) |
| len() | Length | len([1,2,3]) → 3 |
| list() | Convert to list | list((1,2)) → [1,2] |
| locals() | Local variables dict | locals() |
| map() | Apply function | list(map(lambda x:x\*2,[1,2])) → [2,4] |
| max() | Maximum | max([1,2,3]) → 3 |
| min() | Minimum | min([1,2,3]) → 1 |
| next() | Next item of iterator | next(iter([1,2])) → 1 |
| object() | Base object | object() |
| oct() | Convert to octal | oct(8) → '0o10' |
| open() | Open file | open('file.txt','r') |
| ord() | Char → Unicode | ord('A') → 65 |
| pow() | Power | pow(2,3) → 8 |
| print() | Print output | print('hi') |
| property() | Create property | class A: x = property() |
| range() | Number sequence | list(range(3)) → [0,1,2] |
| repr() | String representation | repr(3.14) → '3.14' |
| reversed() | Reverse iterable | list(reversed([1,2])) → [2,1] |
| round() | Round number | round(3.14159,2) → 3.14 |
| set() | Create set | set([1,1,2]) → {1,2} |
| setattr() | Set attribute | setattr(obj,'attr',5) |
| slice() | Slice object | slice(0,3) |
| sorted() | Sort iterable | sorted([3,1]) → [1,3] |
| staticmethod() | Create static method | class A: @staticmethod def f(): pass |
| str() | Convert to string | str(123) → '123' |
| sum() | Sum elements | sum([1,2,3]) → 6 |
| super() | Call parent class | super().method() |
| tuple() | Convert to tuple | tuple([1,2]) → (1,2) |
| type() | Type of object | type(3) → <class 'int'> |
| vars() | Object \_\_dict\_\_ | vars(obj) |
| zip() | Combine iterables | list(zip([1,2],[3,4])) → [(1,3),(2,4)] |
| \_\_import\_\_() | Import module | math = \_\_import\_\_('math') |

This **covers all standard Python 3.x built-in functions** with examples you can copy and try directly.