Python Dictionary — 50 Examples for Extracting Keys, Values, and Items

A complete Light■Theme Illustrated Guide to master Python dict key and item extraction patterns, from basic to advanced.

Method	Purpose	Returns
dict.keys()	Dynamic view of all keys	dict_keys
dict.values()	Dynamic view of values	dict_values
dict.items()	Dynamic view of (key,value) pairs	dict_items
list(d.keys())	Snapshot list of keys	list
zip(*d.items())	Unzip keys and values	tuple of tuples

1. Extract all keys

```
d = {'a':1,'b':2,'c':3}
print(d.keys())
```

2. Convert keys to list

```
keys_list = list(d.keys())
```

3. Iterate over keys

```
for k in d:
    print(k)
```

4. Iterate over keys and values

```
for k in d:
    print(k, d[k])
```

5. Iterate using items()

```
for k,v in d.items():
    print(f"{k} → {v}")
```

6. Extract values only

```
vals = list(d.values())
```

7. Check if key exists

```
if 'a' in d: print('found')
```

8. Use get() safely

```
print(d.get('x', 0))  # default if missing
```

9. Unpack keys & values

```
keys, vals = zip(*d.items())
```

10. Convert to list of tuples

```
pairs = list(d.items())
```

11. Filter by value

```
{k:v for k,v in d.items() if v>1}
```

12. Extract even values

```
[v for v in d.values() if v%2==0]
```

13. Keys starting with 'a'

```
{k for k in d if k.startswith('a')}
```

14. Reverse lookup by value

```
key = next(k for k,v in d.items() if v==2)
```

15. Filter by list of keys

```
wanted={'a','c'}
{k:d[k] for k in wanted if k in d}
```

16. Filter by value type

```
d2={'a':1,'b':'two','c':3}
{k:v for k,v in d2.items() if isinstance(v,int)}
```

17. Top-N largest values

```
sorted(d, key=d.get, reverse=True)[:2]
```

18. Remove None values

```
{k:v for k,v in d.items() if v is not None}
```

19. Subset of keys

```
sub={k:d[k] for k in ['a','b'] if k in d}
```

20. Invert dict

```
inv={v:k for k,v in d.items()}
```

21. Dynamic key view update

```
keys=d.keys(); d['d']=4; print(keys)
```

22. Set operations on keys

```
d1={'a':1,'b':2}; d2={'b':3,'c':4}
print(d1.keys() & d2.keys())
```

23. Iterate sorted by key

```
for k in sorted(d): print(k,d[k])
```

24. Iterate sorted by value

```
for k in sorted(d, key=d.get): print(k,d[k])
```

25. First and last key

```
first=next(iter(d)); last=next(reversed(d))
```

26. Random key selection

```
import random; print(random.choice(list(d.keys())))
```

27. Merge dicts

```
merged = d | {'d':4}  # Python 3.9+
```

28. Numeric-only keys

```
d3={1:'a',2:'b','x':'y'}
[k for k in d3 if isinstance(k,int)]
```

29. Comprehension transformation

```
{k:v**2 for k,v in d.items()}
```

30. Build from zip

```
keys=['x','y']; vals=[10,20]
print(dict(zip(keys,vals)))
```

31. Keys to tuple

```
tuple(d.keys())
```

32. Join keys to string

```
','.join(d.keys())
```

33. Map function over values

```
{k:v*10 for k,v in d.items()}
```

34. Swap keys/values

```
{v:k for k,v in d.items()}
```

35. Filter with lambda

```
dict(filter(lambda kv: kv[1]>1, d.items()))
```

36. Sort by value desc

```
dict(sorted(d.items(), key=lambda kv: kv[1], reverse=True))
```

37. Key with max value

```
max(d, key=d.get)
```

38. Key with min value

```
min(d, key=d.get)
```

39. All keys with a value

```
[k for k,v in d.items() if v==2]
```

40. Unzip to lists

```
keys, vals = zip(*d.items())
```

41. Keys from nested dict

```
nested={'user':{'name':'Alice','age':25}}
print(nested['user'].keys())
```

42. Safe nested access

```
nested.get('user',{}).get('name')
```

43. Flatten one level

```
{f"{k1}_{k2}":v2 for k1,v1 in nested.items() for k2,v2 in v1.items()}
```

44. All keys recursively

```
def all_keys(d):
    for k,v in d.items():
        yield k
        if isinstance(v,dict): yield from all_keys(v)
```

45. Subset by condition

```
{x:y for x,y in d.items() if y%2==1}
```

46. Create fromkeys

```
dict.fromkeys(['x','y','z'],0)
```

47. Merge list of dicts

```
dicts=[{'a':1},{'b':2},{'a':3}]
{k for d in dicts for k in d}
```

48. Unique values across dicts

```
{v for d in dicts for v in d.values()}
```

49. Count key frequencies

```
from collections import Counter
cnt=Counter(k for d in dicts for k in d)
```

50. Pretty print JSON

```
import json
print(json.dumps(d, indent=2))
```

■ Summary

This illustrated guide covered 50 progressively complex examples of extracting keys, values, and items in Python dictionaries.