<k/>

PYTHON BASICS

The things you reach for every day

THEKHANSA.COM

TYPES AND CONVERSION

```
x = 5          # int
y = 2.5        # float
s = "hi"       # str
ok = True      # bool
n = None       # no value
type(x)  int("5")  float("2.5")  str(10)
```

STRINGS

```
name = "Ada"
f"Hello {name}"        # f-string
s.lower()  s.upper()  s.strip()
s.split(",")   ",".join(parts)
s.replace("a", "b")
s[1:4]   s[-1]   len(s)
"py" in s    s.startswith("P")
```

LISTS

```
a = [3, 1, 2]
a.append(4)   a.extend([5, 6])
a.insert(0, 9)   a.pop()   a.remove(1)
a.sort()        # in place
sorted(a, reverse=True)  # new list
a[1:3]   a[::-1]   len(a)
[x * 2 for x in a if x > 1]
```

DICTIONARIES

```
d = {"a": 1, "b": 2}
d["a"]          d.get("z", 0)
d["c"] = 3     del d["a"]
for k, v in d.items(): ...
d.keys()  d.values()
{k: v * 2 for k, v in d.items()}
```

TUPLES AND SETS

```
t = (1, 2)       # cannot change
a, b = t          # unpack
s = {1, 2, 3}     # no duplicates
s.add(4)
s | s2   # union
s & s2   # intersection
s - s2   # difference
```

CONTROL FLOW

```
if x > 0:
    ...
elif x == 0:
    ...
else:
    ...
for i in range(5): ...
for i, v in enumerate(a): ...
for p, q in zip(a, b): ...
while cond: ...   # break / continue
```

FUNCTIONS

```
def add(a, b=0):
    return a + b

def f(*args, **kwargs): ...
square = lambda x: x * x
def f(x: int) -> int: ...  # hints
```

FILES AND ERRORS

```
with open("f.txt") as f:
    text = f.read()
with open("o.txt", "w") as f:
    f.write("hello")

try:
    risky()
except ValueError as e:
    print(e)
finally:
    cleanup()
```

CLASSES

```
class Dog:
    def __init__(self, name):
        self.name = name
    def bark(self):
        return f"{self.name} says woof"

d = Dog("Rex")   d.bark()
```

HANDY BUILT-INS

```
len  sum  min  max  abs  round
sorted  reversed  any  all
range(start, stop, step)
import math, random, datetime
math.sqrt(9)   random.choice(a)
help(obj)   dir(obj)
```

Good to know: Indentation is part of Python: use 4 spaces. Read errors from the bottom up, the last line tells you what went wrong.