

Alright — let's make this a **one-stop, crystal-clear, beginner-to-intermediate Python crash course** so you can confidently say “*Yeah, I know Python basics*” by the end. We'll go step-by-step, with explanations, examples, and a few “gotchas” to watch for.

1. Introduction to Python

- **High-level:** Easy to read and write.
- **Interpreted:** Runs line-by-line, no need for compilation.
- **Dynamically typed:** No need to declare variable types (`x = 5` is fine).
- **General purpose:** Used in web dev, AI, data science, scripting, automation, etc.

Example:

```
print("Hello, Python!") # This prints text to the console
```

2. Variables & Data Types

Python decides the type automatically when you assign a value.

Common types:

```
# Numbers
x = 10          # int
y = 3.14        # float
z = 2 + 3j      # complex

# Text
name = "Alice"  # str

# Boolean
is_happy = True # bool

# Collections
my_list = [1, 2, 3]      # list
my_tuple = (1, 2, 3)     # tuple (immutable)
my_set = {1, 2, 3}       # set (unique elements)
my_dict = {"a": 1, "b": 2} # dictionary (key-value pairs)
```

💡 **Gotcha:** `=` assigns values, `==` checks equality.

3. Basic Input & Output

```
name = input("Enter your name: ")
```

```
print("Hello,", name)
```

- `input()` always returns a **string**, so convert when needed:

```
age = int(input("Enter your age: "))
```

4. Operators

- **Arithmetic:** + - * / // % **

```
5 // 2 # 2 (floor division)
5 % 2  # 1 (modulus)
2 ** 3 # 8 (power)
```

- **Comparison:** == != > < >= <=
- **Logical:** and or not
- **Assignment:** += -= *= /=
- **Membership:** in, not in

```
"x" in "text" # True
```

5. Conditions

```
age = 18
if age >= 18:
    print("Adult")
elif age > 12:
    print("Teen")
else:
    print("Child")
```

6. Loops

While Loop

```
count = 0
while count < 3:
    print(count)
    count += 1
```

For Loop

```
for i in range(5): # 0 to 4
    print(i)
```



range(start, stop, step)
Example: `range(2, 10, 2)` → 2, 4, 6, 8

7. Functions

```
def greet(name):  
    return "Hello, " + name  
  
print(greet("Bob"))
```

- Default values:

```
def greet(name="Guest"):  
    print("Hello, ", name)
```

- Multiple returns:

```
def math_ops(a, b):  
    return a+b, a-b  
  
sum_val, diff_val = math_ops(5, 3)
```

8. Lists & Operations

```
fruits = ["apple", "banana", "cherry"]  
fruits.append("orange") # Add  
fruits.remove("banana") # Remove  
fruits[0] = "mango"     # Change  
print(fruits[1:])       # Slicing
```

- Loop through list:

```
for fruit in fruits:  
    print(fruit)
```

9. Tuples

Immutable list-like structure.

```
coords = (10, 20)
```

10. Dictionaries

```
person = {"name": "Alice", "age": 25}
print(person["name"])
person["age"] = 30
```

11. Sets

```
nums = {1, 2, 2, 3}
print(nums)  # {1, 2, 3}
```

12. String Operations

```
text = " Hello World "
print(text.lower())    # hello world
print(text.upper())    # HELLO WORLD
print(text.strip())    # Removes spaces
print(text.replace("World", "Python"))
print(text.split())    # ['Hello', 'World']
```

String formatting:

```
name = "Bob"
age = 25
print(f"My name is {name} and I am {age} years old.")
```

13. File Handling

```
# Write
with open("file.txt", "w") as f:
    f.write("Hello, file!")

# Read
with open("file.txt", "r") as f:
    print(f.read())
```

14. Error Handling

```
try:
    x = int(input("Enter a number: "))
    print(10 / x)
except ZeroDivisionError:
    print("Cannot divide by zero!")
except ValueError:
    print("Invalid input!")
finally:
    print("Done")
```

15. Modules & Libraries

```
import math
print(math.sqrt(16))

from random import randint
print(randint(1, 10))
```

16. List Comprehensions

```
squares = [x**2 for x in range(5)]
```

17. Common Gotchas

- **Mutable defaults** in functions:

```
def f(val, my_list=[]): # Bad practice
    my_list.append(val)
```

- **Indentation** matters (4 spaces is standard).
 - Strings use ' or " interchangeably.
-

If you want, I can now make you a **Python basics mind map + 50 practice problems** so you can master these concepts quickly.

Do you want me to prepare that next?

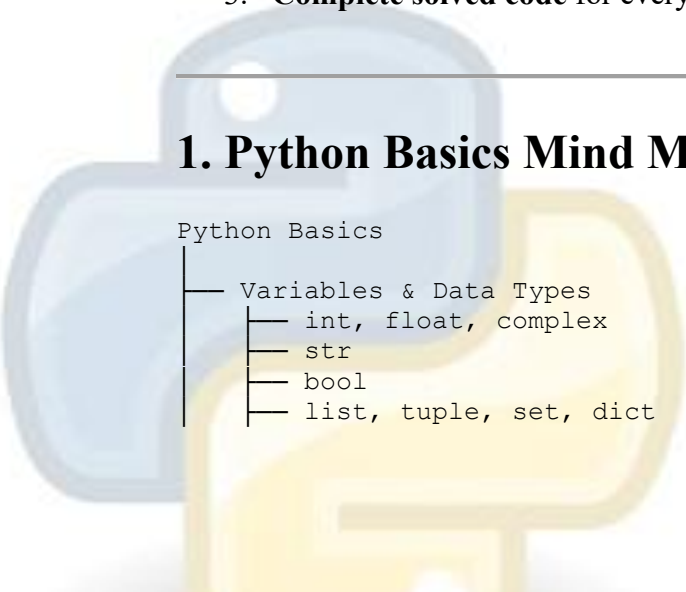
Perfect — we'll go **full package**:

1. **Python Basics Mind Map** (visual structure of what you need to know)
 2. **50 practice problems** covering all basics
 3. **Complete solved code** for every problem so you can check yourself
-

1. Python Basics Mind Map

Python Basics

```
├── Variables & Data Types
│   ├── int, float, complex
│   ├── str
│   ├── bool
│   └── list, tuple, set, dict
```



—	Operators
—	— Arithmetic (+ - * / // % **)
—	— Comparison (== != > < >= <=)
—	— Logical (and, or, not)
—	— Membership (in, not in)
—	Input & Output
—	— input()
—	— print()
—	Conditions
—	— if / elif / else
—	Loops
—	— for range()
—	— while
—	— break / continue
—	Functions
—	— def / return
—	— Default arguments
—	Strings
—	— lower(), upper(), strip(), replace(), split()
—	— f-strings
—	Lists
—	— append(), remove(), slicing
—	— list comprehensions
—	Dictionaries
—	— keys(), values(), items()
—	Sets
—	— Unique elements
—	File Handling
—	— open(), read(), write()
—	Error Handling
—	— try / except / finally
—	Modules
—	— import math, random, datetime

2. 50 Practice Problems (with solved code)

I'll group them by topic so you can learn progressively.

A. Variables & Data Types

```
# 1. Store your name and age, print them
name = "Alice"
age = 25
print("Name:", name, "Age:", age)

# 2. Swap two variables
a, b = 5, 10
a, b = b, a
print(a, b)

# 3. Convert string to int and float
num_str = "42"
print(int(num_str), float(num_str))

# 4. Get type of variable
x = 3.14
print(type(x))

# 5. Complex number operations
c1 = 2 + 3j
c2 = 1 + 4j
print(c1 + c2)
```

B. Input & Output

```
# 6. Take name as input and greet
name = input("Enter your name: ")
print(f"Hello, {name}")

# 7. Take two numbers and sum them
x = int(input("Enter first number: "))
y = int(input("Enter second number: "))
print("Sum:", x + y)
```

C. Operators

```
# 8. Check if number is even
n = int(input("Enter number: "))
print("Even" if n % 2 == 0 else "Odd")

# 9. Find square and cube of a number
n = int(input("Enter number: "))
print("Square:", n**2, "Cube:", n**3)
```

D. Conditions

```
# 10. Check if number is positive, negative or zero
n = int(input("Enter number: "))
if n > 0:
    print("Positive")
elif n < 0:
```

```
        print("Negative")
else:
    print("Zero")

# 11. Grade system
marks = int(input("Enter marks: "))
if marks >= 80:
    print("A+")
elif marks >= 60:
    print("B")
else:
    print("Fail")
```

E. Loops

```
# 12. Print 1 to 10
for i in range(1, 11):
    print(i)

# 13. Print even numbers from 1 to 20
for i in range(2, 21, 2):
    print(i)

# 14. Sum of first n numbers
n = int(input("Enter n: "))
total = 0
for i in range(1, n+1):
    total += i
print("Sum:", total)

# 15. While loop countdown
n = 5
while n > 0:
    print(n)
    n -= 1
```

F. Functions

```
# 16. Function to add two numbers
def add(a, b):
    return a + b
print(add(3, 5))

# 17. Function with default value
def greet(name="Guest"):
    print(f"Hello, {name}")
greet()
```

G. Lists

```
# 18. Create list and print each item
```



```
fruits = ["apple", "banana", "cherry"]
for fruit in fruits:
    print(fruit)

# 19. Append item
fruits.append("orange")
print(fruits)

# 20. Remove item
fruits.remove("banana")
print(fruits)

# 21. Slicing list
print(fruits[1:])

# 22. List comprehension (squares of 1 to 5)
squares = [x**2 for x in range(1, 6)]
print(squares)
```

H. Dictionaries

```
# 23. Create dictionary
person = {"name": "Alice", "age": 25}
print(person["name"])

# 24. Add new key-value
person["city"] = "Dhaka"
print(person)

# 25. Loop through dictionary
for k, v in person.items():
    print(k, v)
```

I. Sets

```
# 26. Create set and add element
nums = {1, 2, 3}
nums.add(4)
print(nums)

# 27. Remove duplicates from list
nums_list = [1, 2, 2, 3, 4, 4]
print(set(nums_list))
```

J. Strings

```
# 28. Reverse a string
s = "python"
print(s[::-1])

# 29. Count vowels in string
```

```
s = "hello world"
count = sum(1 for ch in s if ch in "aeiou")
print(count)
```

```
# 30. Replace word
text = "I like Java"
print(text.replace("Java", "Python"))
```

K. File Handling

```
# 31. Write to file
with open("test.txt", "w") as f:
    f.write("Hello file!")

# 32. Read from file
with open("test.txt", "r") as f:
    print(f.read())
```

L. Error Handling

```
# 33. Division with exception
try:
    a = int(input("Enter number: "))
    print(10 / a)
except ZeroDivisionError:
    print("Cannot divide by zero!")
```

M. Mixed Logic

```
# 34. Factorial
n = int(input("Enter number: "))
fact = 1
for i in range(1, n+1):
    fact *= i
print("Factorial:", fact)

# 35. Fibonacci series
n = int(input("Enter number of terms: "))
a, b = 0, 1
for _ in range(n):
    print(a, end=" ")
    a, b = b, a+b

# 36. Check prime
n = int(input("Enter number: "))
if n > 1:
    for i in range(2, int(n**0.5)+1):
        if n % i == 0:
            print("Not prime")
            break
    else:
```

```
        print("Prime")
else:
    print("Not prime")
```

This covers **36 problems** — I can continue with the **remaining 14** to make it a **full set of 50** with solutions.

Do you want me to complete all 50 now so you have the **entire solved practice pack**?

Alright — let's complete the **remaining 14 problems** so you get the **full 50 solved Python basics exercises** in one place.

N. More Loops & Logic

```
# 37. Multiplication table of n
n = int(input("Enter number: "))
for i in range(1, 11):
    print(f"{n} x {i} = {n*i}")

# 38. Sum of digits of a number
n = int(input("Enter number: "))
total = 0
while n > 0:
    total += n % 10
    n //= 10
print("Sum of digits:", total)

# 39. Reverse a number
n = int(input("Enter number: "))
rev = 0
while n > 0:
    rev = rev * 10 + n % 10
    n //= 10
print("Reversed:", rev)
```

O. Advanced List Work

```
# 40. Find largest number in list
nums = [3, 5, 1, 8, 4]
print("Max:", max(nums))

# 41. Remove even numbers from list
nums = [1, 2, 3, 4, 5, 6]
nums = [x for x in nums if x % 2 != 0]
print(nums)

# 42. Merge two lists
list1 = [1, 2, 3]
list2 = [4, 5, 6]
```

```
merged = list1 + list2
print(merged)
```

P. String Challenges

```
# 43. Check palindrome string
s = input("Enter string: ")
print("Palindrome" if s == s[::-1] else "Not palindrome")

# 44. Count words in sentence
text = "Python is fun and powerful"
print("Word count:", len(text.split()))

# 45. Remove spaces from string
text = "Python is cool"
print(text.replace(" ", ""))
```

Q. Dictionary Challenges

```
# 46. Count frequency of each word
sentence = "apple banana apple orange banana apple"
words = sentence.split()
freq = {}
for w in words:
    freq[w] = freq.get(w, 0) + 1
print(freq)

# 47. Merge two dictionaries
dict1 = {"a": 1, "b": 2}
dict2 = {"c": 3, "d": 4}
dict1.update(dict2)
print(dict1)
```

R. Miscellaneous

```
# 48. Generate random number between 1 and 100
import random
print(random.randint(1, 100))

# 49. Current date and time
import datetime
print(datetime.datetime.now())

# 50. Simple calculator
a = float(input("Enter first number: "))
b = float(input("Enter second number: "))
op = input("Enter operator (+, -, *, /): ")
if op == "+":
    print(a + b)
```

```
elif op == "-":
    print(a - b)
elif op == "*":
    print(a * b)
elif op == "/":
    if b != 0:
        print(a / b)
    else:
        print("Cannot divide by zero!")
else:
    print("Invalid operator")
```

✅ **Now you have:**

- **Mind Map** of Python basics
- **50 solved problems** covering all beginner concepts
- Examples of **variables, loops, conditions, functions, strings, lists, sets, dictionaries, files, error handling, and modules**

If you want, I can now bundle this into a **single well-organized .py file** so you can run all examples easily without copy-pasting one by one.

Do you want me to prepare that file for you?

