

Getting Started with Python

Basic Concepts and Code Examples for Beginners

1. Hello, World!

This is the first Python program most beginners write. It prints text to the screen:

```
print("Hello, World!")
```

When you run it, you'll see:

```
Hello, World!
```

2. Variables and Types

A variable stores a value you can use later:

```
x = 5          # integer
name = "Sam"   # string
pi = 3.14      # float
```

Python figures out the type automatically. For example, here `x` is an integer, and `name` is text (a string).

3. Lists

A list is a collection of values:

```
numbers = [1, 2, 3, 4]
print(numbers)
```

You can add lists together:

```
a = [1, 2]
b = [3, 4]
print(a + b)    # [1, 2, 3, 4]
```

Lists can contain any type of item.

4. Basic Operators

Python lets you do math and combine values:

```
print(2 + 3)    # addition
print(5 * 2)    # multiplication
print(10 % 3)   # remainder
```

You can also use operators with strings and lists:

```
print("hi" * 3)      # hi repeated 3 times
print([1] * 4)        # [1, 1, 1, 1]
```

Adding lists joins them.

5. String Formatting

You can put variables into text easily:

```
name = "Alex"
age = 20
print(f"{name} is {age} years old.")
```

This prints the values inside a sentence using an f-string.

6. Basic String Operations

Strings are text, and you can work with them:

```
s = "hello"
print(s.upper())    # HELLO
print(s[1:4])       # "ell" (slice)
```

Strings support indexing and many built-in methods.

7. Conditions

Use if, elif, and else to make decisions:

```
x = 10

if x > 5:
    print("x is big")
elif x == 5:
    print("x is five")
```

```
else:  
    print("x is small")
```

Python checks each condition top to bottom and runs the first true block.

8. Loops

Loops let you repeat code. For loop — for each item in something:

```
for i in range(5):  
    print(i)
```

While loop — repeat while a condition is true:

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

Loops are great for repeating tasks.