

Chapter 8 - 2Var Datatypes Identifiers

Theory:

This chapter covers the topic of 2Var Datatypes Identifiers.

Code Example:

```
# -----
# VARIABLES AND IDENTIFIERS
# -----
# Variable Declaration
a = 1
b = 2
# To evaluate expressions inside curly braces {}, use f-strings by placing 'f' before the
string
print(f"{a} + {b} = {a + b}") # Output: 1 + 2 = 3
# -----
# IDENTIFIERS
# -----
# Identifiers are names used for variables, functions, classes, etc.
# Identifiers are case-sensitive
Name = 'Milan' # Capital 'N'
name = 'Milan' # Small 'n'
# 'Name' and 'name' are two different identifiers
# -----
# DATA TYPES
# -----
a = 1 # Integer (int)
b = 5.33 # Floating-point number (float)
c = "Milan" # String (str)
d = False # Boolean (bool)
e = True # Boolean (bool)
f = None # NoneType (represents 'nothing' or 'no value')
# -----
# IDENTIFIER RULES
# -----
# ■ Valid variable names:
# - Must begin with a letter (a-z or A-Z) or an underscore (_)
# - Can contain letters, digits (0-9), and underscores
# - Are case-sensitive (myVar ≠ MyVar)
# ■ Invalid variable names:
# - Cannot start with a number (e.g., lname ■)
# - Cannot include spaces (e.g., my var ■)
# - Cannot use special symbols like @, $, %, etc.
# ■ Examples of valid identifiers:
my_variable = 10
_myVar = "hello"
name1 = "Milan"
# ■ Examples of invalid identifiers (will cause SyntaxError if used):
# lname = "Invalid"
# my var = "Invalid"
# @name = "Invalid"
# -----
# OPERATORS IN PYTHON
# -----
```

```

# -----
# 1. Arithmetic Operators
# -----
a = 7
b = 5
print("a + b =", a + b) # Addition
print("a - b =", a - b) # Subtraction
print("a * b =", a * b) # Multiplication
print("a / b =", a / b) # Division (float)
print("a % b =", a % b) # Modulus (remainder)
# -----
# 2. Assignment Operators
# -----
a = 7 - 2 # Assigns 5 to a
b = 5
b += 3 # b = b + 3 → b = 8
b -= 3 # b = b - 3 → b = 5
print("Value of a:", a)
print("Final value of b:", b)
# -----
# 3. Comparison Operators
# (Returns True or False)
# -----
print("5 < 4 =", 5 < 4) # False
print("5 >= 4 =", 5 >= 4) # True
print("5 <= 4 =", 5 <= 4) # False
print("5 != 4 =", 5 != 4) # True
print("5 == 4 =", 5 == 4) # False
print("5 > 4 =", 5 > 4) # True
# -----
# 4. Logical Operators
# (Used with boolean values)
# -----
e = True or False
print("True OR False =", e)
e = True and False
print("True AND False =", e)
e = not True
print("NOT True =", e)
# -----
# TRUTH TABLES
# -----
# OR Truth Table (A or B)
# A B A or B
# -----
# False False False
# False True True
# True False True
# True True True
# AND Truth Table (A and B)
# A B A and B
# -----
# False False False

```

```

# False True False
# True False False
# True True True
# NOT Truth Table (not A)
# A not A
# -----
# True False
# False True
# -----
# TYPES OF VARIABLES
# -----
a = 1 # Integer (int)
b = 5.33 # Floating-point number (float)
c = "Milan" # String (str)
d = False # Boolean (bool)
e = True # Boolean (bool)
# -----
# FINDING THE TYPE OF A VARIABLE
# -----
t = type(a) #
print("Type of a:", t)
t = type(b) #
print("Type of b:", t)
t = type(c) #
print("Type of c:", t)
t = type(d) #
print("Type of d:", t)
t = type(e) #
print("Type of e:", t)
# -----
# TYPE CONVERSION (CASTING)
# -----
a = "31.2" # String type
print("Before conversion, type of a:", type(a))
b = float(a) # Convert string to float
print("After conversion, type of b:", type(b))
# -----
# INPUT FUNCTION
# -----
# Taking input as STRING by default
a = input("Enter number 1: ") # Example: "5"
b = input("Enter number 2: ") # Example: "5"
print("You entered (as strings):", a, b)
print("String Addition (a + b):", a + b) # "5" + "5" = "55"
# -----
# INPUT WITH TYPE CONVERSION
# -----
# Convert user input to integer for numerical addition
c = int(input("Enter number 1: ")) # Example: 5
f = int(input("Enter number 2: ")) # Example: 5
print("You entered (as integers):", c, f)
print("Numeric Addition (c + f):", c + f) # 5 + 5 = 10

```