PYTHON IDENTIFIER

- An identifier is the name you give to any variable, function, class, module, or object in your program.
- An identifier must follow certain rules in Python, and these rules apply to all kinds of identifiers, whether they are variables, functions, classes, or other objects.

Characteristics of Identifiers→

1. Alphanumeric Characters:
   - Identifiers can contain letters (a-z, A-Z), digits (0-9), and underscores (_).
   - However, they cannot start with a digit.
2. Case Sensitive:
   - Python is case-sensitive. This means that myVariable, MyVariable, and MYVARIABLE are considered different identifiers.
3. Cannot be Python Keywords:
   - Identifiers cannot be one of Python's reserved keywords (e.g., if, else, while, class, etc.).
4. Can be Descriptive:
   - Identifiers should be meaningful and descriptive to make the code more readable.
5. Special Characters:
   - Identifiers cannot contain spaces or special characters like @, #, $, etc. Only underscores _ are allowed to separate words (e.g., my_variable).

Rules for Naming Identifiers→

1. Starts with a letter or underscore:
   - my_var is valid.
   - _myVar is valid.
   - 123abc is invalid (because it starts with a number).

2. Can contain letters, digits, and underscores:
   - my_variable, var_1, a2b3 are valid.

- my-variable is invalid (hyphen is not allowed).

3. Cannot be a keyword:
   - if, else, try, class are invalid as identifiers because they are reserved keywords in Python.
4. Cannot contain spaces:
   - my variable is invalid.

Valid Identifiers:

```python
age = 25                # valid
name = "John"           # valid
my_var = 100            # valid
score_1 = 89            # valid
_PI = 3.14159           # valid (underscore at the beginning)
total_amount = 150.75   # valid
myVariable = 10         # valid (case-sensitive)
```

Invalid Identifiers:

```python
1st_place = "First"     # invalid (starts with a number)
total-price = 200       # invalid (contains a hyphen)
class = "Python"        # invalid (reserved keyword)
my variable = "space"   # invalid (contains a space)
```

Python Reserved Keywords→
These are reserved words that cannot be used as identifiers because Python uses them for special purposes.

```python
import keyword
print(keyword.kwlist)  # List all Python reserved keywords
```

COMPLETE IDENTIFIER IN PYTHON →

Variable Identifier:

```python
x = 10       # 'x' is a valid identifier that holds the value 10
y = "Alice"  # 'y' is a valid identifier holding the string "Alice"
```

Function Identifier:

```python
def greet(name):  # 'greet' is a valid function identifier
    print(f"Hello, {name}!")

greet("Bob")  # Output: Hello, Bob!
```

Class Identifier:

```python
class Car:  # 'Car' is a valid class identifier
    def __init__(self, make, model):
        self.make = make
        self.model = model

my_car = Car("Toyota", "Corolla")  # 'my_car' is an identifier for the Car
```

Module Identifier:

```python
import math  # 'math' is the name of the module, and it's an identifier
print(math.pi)  # Accessing 'pi' from the 'math' module
```

- Identifiers are names used to identify variables, functions, classes, modules, and other objects in Python.

- They must follow specific naming rules (e.g., not starting with a number, being case-sensitive, not being a reserved keyword).
- Good naming practices make your code more readable and maintainable, helping both you and others understand the program's intent.