Identifiers and Reserved Words in Python

Identifiers in Python

Identifiers are names used to identify variables, functions, classes, modules, and other objects in Python. Here are some rules and conventions for Python identifiers:

1. Naming Rules:

- Identifiers can be a combination of letters in lowercase (`a` to `z`) or uppercase (`A` to `Z`) or digits (`0` to `9`) or an underscore (`_`).
- An identifier cannot start with a digit.
- Python identifiers are case-sensitive (`Variable` and `variable` are different).
- Reserved words (keywords) cannot be used as identifiers.

2. Best Practices:

- Use meaningful names to make the code more readable.
- Use underscores to separate words in a variable name (snake_case).
- For class names, use CamelCase (e.g., `MyClass`).
- Constants are usually written in all uppercase letters with underscores separating words (e.g., `MAX_SIZE`).

```
# Valid Identifiers
my_variable = 10
MyVariable = 20
my_variable_2 = 30

# Invalid Identifiers
2variable = 40  # Cannot start with a digit
my-variable = 50  # Hyphens are not allowed
```

Reserved Words in Python

Python has a set of reserved words or keywords that have special meanings and cannot be used as identifiers. Here is a list of the reserved words in Python:

```
['False', 'None', 'True', 'and', 'as', 'assert', 'async', 'await', 'break', 'class', 'contir
```

Usage Examples

```
# Using reserved words in context
def example_function():
    try:
        for i in range(5):
            if i % 2 == 0:
```

```
                print(f"{i} is even")
            else:
                print(f"{i} is odd")
    except Exception as e:
        print(f"An error occurred: {e}")

example_function()
```

Stay Updated

Be sure to this repository to stay updated with new examples and enhancements!

License

This project is protected under the MIT License.

Contact

Panagiotis Moschos - pan.moschos86@gmail.com

Note: This is a Python script and requires a Python interpreter to run.

Happy Coding

Made with by Panagiotis Moschos (https://github.com/pmoschos)