

Έννοιες Γραμμής Εντολών στην Python

Python Command Line Concepts

The `if __name__ == '__main__'` Pattern

Το Μοτίβο `if __name__ == '__main__'`

📖 Όταν εκτελούμε ένα αρχείο Python, η μεταβλητή `__name__` παίρνει διαφορετικές τιμές ανάλογα με τον τρόπο εκτέλεσης:

- Αν το αρχείο εκτελείται απευθείας, το `__name__` γίνεται `'__main__'`
- Αν το αρχείο εισάγεται ως module, το `__name__` γίνεται το όνομα του module

🇬🇧 When we run a Python file, the `__name__` variable takes different values depending on how it's executed:

- If the file is run directly, `__name__` becomes `'__main__'`
- If the file is imported as a module, `__name__` becomes the module's name

```
# example.py
def calculate_square(n):
    return n * n

def main():
    # Κύρια λογική του προγράμματος
    # Main program logic
    number = 5
    result = calculate_square(number)
    print(f"The square of {number} is {result}")

if __name__ == '__main__':
    main()
```

Command Line Arguments

Ορίσματα Γραμμής Εντολών

📖 Το module `sys` παρέχει πρόσβαση στα ορίσματα γραμμής εντολών μέσω της λίστας `sys.argv`:

- `sys.argv[0]`: Το όνομα του script
- `sys.argv[1:]`: Τα ορίσματα που παρέχονται

🇬🇧 The `sys` module provides access to command-line arguments through the `sys.argv` list:

- `sys.argv[0]`: The script name
- `sys.argv[1:]`: The provided arguments

```
# cli_args.py
import sys

def process_arguments():
    # Έλεγχος αν δόθηκαν ορίσματα
```

```

# Check if arguments were provided
if len(sys.argv) < 2:
    print("Usage: python cli_args.py <number>")
    sys.exit(1)

try:
    number = int(sys.argv[1])
    result = number * number
    print(f"The square of {number} is {result}")
except ValueError:
    print("Error: Please provide a valid number")
    sys.exit(1)

if __name__ == '__main__':
    process_arguments()

```

Python Path and System Information

Python Path και Πληροφορίες Συστήματος

📖 Το `sys.path` είναι μια λίστα που περιέχει τις τοποθεσίες όπου η Python αναζητά modules:

🇬🇧 `sys.path` is a list containing locations where Python looks for modules:

```

# path_info.py
import sys

def show_python_info():
    # Εμφάνιση της τρέχουσας έκδοσης Python
    # Display current Python version
    print(f"Python version: {sys.version}")

    # Εμφάνιση του Python Path
    # Display Python Path
    print("\nPython Path:")
    for path in sys.path:
        print(f"- {path}")

if __name__ == '__main__':
    show_python_info()

```

Running Examples from Terminal

Εκτέλεση Παραδειγμάτων από το Τερματικό

📖 Για να εκτελέσετε τα παραπάνω παραδείγματα:

1. `python example.py`
2. `python cli_args.py 10`
3. `python path_info.py`

🇬🇧 To run the above examples:

1. `python example.py`
2. `python cli_args.py 10`
3. `python path_info.py`

Additional Notes / Επιπρόσθετες Σημειώσεις

🇬🇷 Καλές πρακτικές:

- Πάντα να ελέγχετε τα ορίσματα πριν τα χρησιμοποιήσετε
- Να παρέχετε μηνύματα χρήσης όταν λείπουν ορίσματα
- Να χρησιμοποιείτε το `sys.exit()` με κατάλληλους κωδικούς σφάλματος

🇬🇷 Best practices:

- Always validate arguments before using them
- Provide usage messages when arguments are missing
- Use `sys.exit()` with appropriate error codes

In []:

Διαχείριση Πακέτων Python

Python Package Management

🇬🇷 Η Python χρησιμοποιεί το `pip` ως τον προεπιλεγμένο διαχειριστή πακέτων. Το PyPI (Python Package Index - pypi.org) είναι το επίσημο αποθετήριο λογισμικού της Python.

🇬🇷 Python uses `pip` as its default package manager. PyPI (Python Package Index - pypi.org) is Python's official software repository.

Basic pip Commands / Βασικές Εντολές pip

(Εκτέλεση μέσω Terminal/ ΔΕΝ αποτελούν εντολές Python)

```
# Εγκατάσταση πακέτου
# Install a package
pip install package_name

# Εγκατάσταση συγκεκριμένης έκδοσης
# Install specific version
pip install package_name==1.0.0

# Αναβάθμιση πακέτου
# Upgrade a package
pip install --upgrade package_name

# Εμφάνιση εγκατεστημένων πακέτων
# Show installed packages
pip list

# Εξαγωγή λίστας πακέτων
# Export package list
pip freeze > requirements.txt

# Εγκατάσταση από αρχείο requirements
# Install from requirements file
pip install -r requirements.txt

# Απεγκατάσταση πακέτου
# Uninstall a package
pip uninstall package_name
```

In []: