Python

Python Modules and Packages

About the Author

- Created By: Mohammad Salman
- Experience: 19 Years +
- Designation: Corporate Trainer .NET

Microsoft CERTIFIED Professional

Microsoft Specialist

Microsoft CERTIFIED Solutions Developer

Icons Used

Questions

Tools

Hands-on Exercise

Coding Standards

Questions?

Reference

Try it Out

Informative Slide

Mandatory Slide

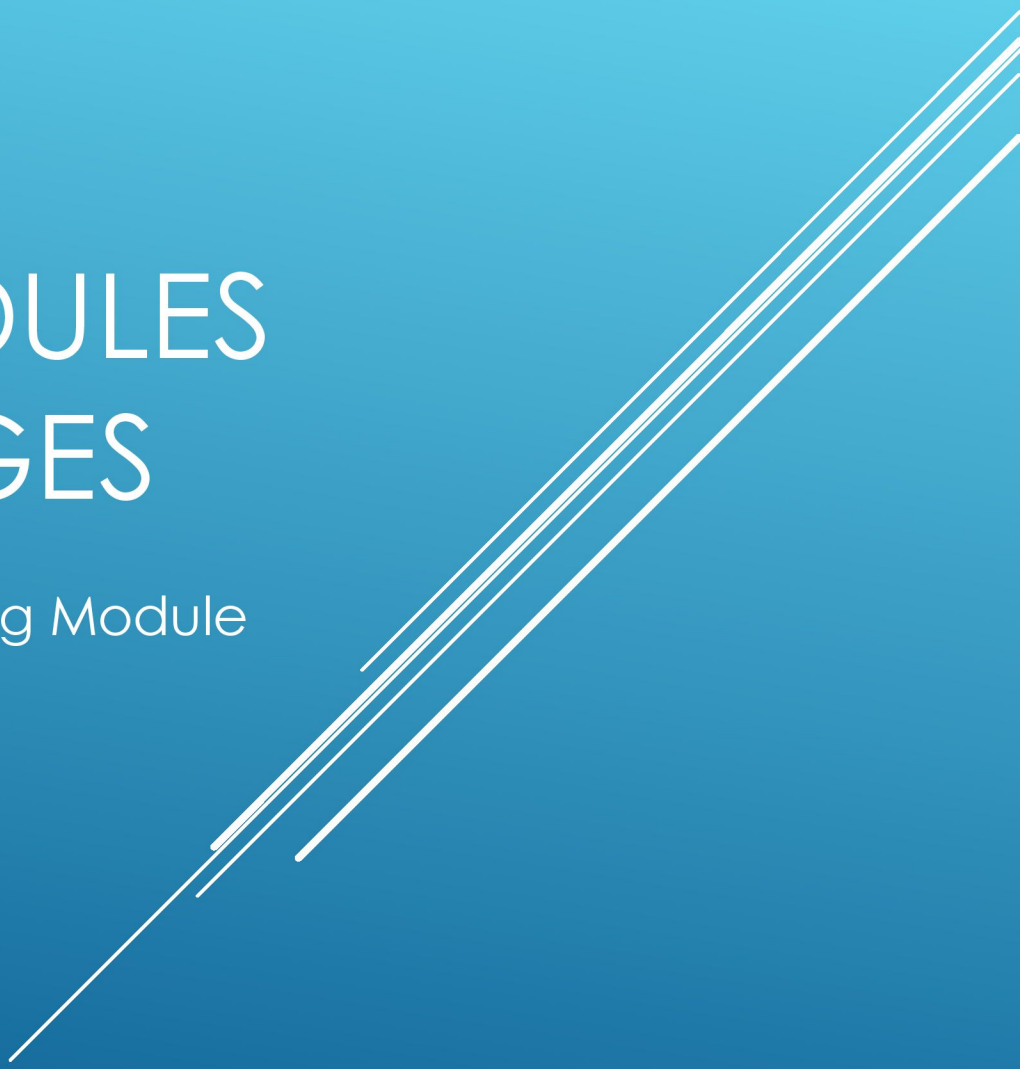
Welcome Break

PYTHON MODULES AND PACKAGES

- Learn Python Modules and Packages
  By: Mohd Salman

PYTHON MODULES AND PACKAGES

Beginner to Advanced Training Module



LEARNING OBJECTIVES

- • Understand modules and packages.
- • Use built-in/custom modules.
- • Write your own modules.
- • Organize code with packages.
- • Apply modular programming.

WHAT IS A MODULE?

- **A module is a Python file (.py) containing reusable code.**

Benefits:

- • Code reuse
- • Debugging ease
- • Maintainability

Example:

```
import math
print(math.sqrt(16))
print(math.pi)
```

USING BUILT-IN MODULES

Common modules:

- • math
- • random
- • datetime

- Example:

```
import random
print(random.randint(1,10))
```

Methods:

import math

from math import sqrt, pi

import math as m

IMPORTING MODULES

WRITING YOUR OWN MODULE

Steps:

1. Create a .py file.
2. Import it.

Example:

calculator.py:

```
def add(a,b): return a+b
```

main.py:

```
import calculator
print(calculator.add(3,5))
```

PACKAGES IN PYTHON

A package is a directory with modules + __init__.py.

Structure:

ecommerce/

  __init__.py

  cart.py

Usage:

from ecommerce import cart

cart.add_item()

__INIT__.PY EXPLAINED

Marks a directory as a package.

Can initialize/expose functions.

Example:

```
from .cart import add_item
```

BEST PRACTICES

- • Descriptive names.
- • Avoid circular imports.
- • Keep modules focused.
- • Add docstrings.

KNOWLEDGE CHECK

- What marks folder as package?
- Valid import?
- Circular imports cause?

KNOWLEDGE CHECK

- Module = single .py file.
- __init__.py marks package.
- Avoid wildcard imports.
- Circular imports = errors.

RECAP

- Modules = reusable files.
- Packages = directories of modules.
- Importing = modular programs.