

Lecture 05

Lect. PhD.
Arthur Molnar

Modular
Programming
Introduction
Python Modules
Python Packages
Modular
programming in
Assignment 4

Modular Programming

Lect. PhD. Arthur Molnar

Babes-Bolyai University

Overview

Lecture 05

Lect. PhD.
Arthur Molnar

Modular Programming

Introduction
Python Modules
Python Packages
Modular
programming in
Assignment 4

1 Modular Programming

- Introduction
- Python Modules
- Python Packages
- Modular programming in Assignment 4

Modules

Lecture 05

Lect. PhD.
Arthur Molnar

Modular Programming

Introduction
Python Modules
Python Packages
Modular
programming in
Assignment 4

Modular programming - a software design technique that increases the extent to which software is composed of independent, interchangeable components called **modules**, each of which does one aspect within the program and contains everything necessary to accomplish this.

Modules

Lecture 05

Lect. PhD.
Arthur Molnar

Modular
Programming

Introduction

Python Modules

Python Packages

Modular
programming in
Assignment 4

Modules are:

- Independent
- Interchangeable

Modules

Lecture 05

Lect. PhD.
Arthur Molnar

Modular
Programming

Introduction

Python Modules

Python Packages

Modular
programming in
Assignment 4

Discussion

Why is modular programming needed? Advantages and drawbacks...

Modules

Lecture 05

Lect. PhD.
Arthur Molnar

Modular
Programming

Introduction
Python Modules
Python Packages
Modular
programming in
Assignment 4

- Allows grouping related functionalities
- Allows easier delivery and deployment of related functionalities
- Helps with solving naming conflicts

Modules in Python

Lecture 05


Lect. PhD.
Arthur Molnar

Modular
Programming

Introduction
Python Modules
Python Packages
Modular
programming in
Assignment 4

A Python module¹ - a file containing Python statements and definitions (executable statements).

- **Name:** The file name is the module name with the suffix ".py" appended
- **Docstring:** triple-quoted module doc string that defines the contents of the module file. Provide summary of the module and a description about the module's contents, purpose and usage.
- **Executable statements:** function definitions, module variables, initialization code

¹<https://docs.python.org/3/tutorial/modules.html> 

Importing modules

Lecture 05

Lect. PhD.
Arthur Molnar

Modular
Programming
Introduction
Python Modules
Python Packages
Modular
programming in
Assignment 4

In order to use a module it must be imported first. The import statement:

- 1 Searches the global namespace for the module. If the module exists, it is already imported and nothing more needs to be done.
- 2 Searches for the module.
- 3 Variables and functions defined in the module are inserted into a new symbol table (a new namespace). Only the module name is added to the current symbol table

Module search path

Lecture 05

Lect. PhD.
Arthur Molnar

Modular Programming

Introduction
Python Modules
Python Packages
Modular
programming in
Assignment 4

Where does the **'import spam'** statement search for module *spam.py*?

- Built-in modules with the given name
- Directories in the *sys.path* variable:
 - Directory containing the input script
 - Directories specified by environment variable **PYTHONPATH**
 - Directories specified by the environment variable **PYTHONHOME**, an installation-dependent default path

If the module name can't be found anywhere, an **ImportError** exception is raised.

Demo

Lecture 05

Lect. PhD.
Arthur Molnar

Modular
Programming

Introduction

Python Modules

Python Packages

Modular
programming in
Assignment 4

Modules

ex09_modules

Demo

Lecture 05

Lect. PhD.
Arthur Molnar

Modular
Programming

Introduction
Python Modules
Python Packages
Modular
programming in
Assignment 4

Environment Variables

This website has more info on accessing and changing environment variables in the Windows OS -
www.computerhope.com/issues/ch000549.htm

Learning more about modules

Lecture 05

Lect. PhD.
Arthur Molnar

Modular
Programming
Introduction
Python Modules
Python Packages
Modular
programming in
Assignment 4

- **dir(module_name)** can be used to examine the module's symbol tables.
- **help(module_name)** can be used to get help on the module, its data types and functions.
- **pydoc** - A module that allows you to save extracted documentation to HTML format. Best used in command line at the operating system prompt.

Packages

Lecture 05

Lect. PhD.
Arthur Molnar

Modular
Programming
Introduction
Python Modules
Python Packages
Modular
programming in
Assignment 4

- Packages² are a way of structuring Python's module namespace by using "dotted module names"
- **A.B** denotes submodule **B** found in package **A**.
- The same rules apply for importing packages as with modules
- On the drive, directory hierarchies represent packages, so **B.py** will be found in a directory called **A**
- Each package directory contains an `__init__.py` file, telling Python to interpret it as a collection of modules
- `__init__.py` can be empty, or include package initialization code.

²<https://docs.python.org/3/tutorial/modules.html#packages> 🔍 🔍 🔍

Required modules for Assignment 4

Lecture 05

Lect. PhD.
Arthur Molnar

Modular
Programming
Introduction
Python Modules
Python Packages
Modular
programming in
Assignment 4

Create modules for:

- **User interface** - Functions related to user interaction. Contains input and data validation, print operations. This is the only module where input/print operations are present.
- **Functions** - Contains functions required to implement program features
- **Start** - Code that starts the program by calling the required UI function(s)

Demo

Lecture 05

Lect. PhD.
Arthur Molnar

Modular Programming

Introduction
Python Modules
Python Packages
Modular
programming in
Assignment 4

Code review

The code in the following archive is a modular implementation of the calculator program for rational numbers:

ex10_modular_calc