Day 1

if...elif...else:

- Used for decision-making in Python.
- Executes different blocks of code based on the conditions specified.

for...in:

- Used to iterate over a sequence (like a list, tuple, or string) or other iterable objects.
- Executes a set of statements for each element in the sequence.

while:

- Used to create a loop that continues to execute as long as a specified condition is true.
- Executes a set of statements as long as the condition is met.

break:

- Used to exit the current loop prematurely.
- Terminates the loop containing it and transfers the execution to the next statement after the loop.

continue:

- Used to skip the rest of the code inside a loop for the current iteration.
- Resumes the loop with the next iteration.

Ternary Operator:

- A concise way to write simple if...else statements in a single line.
- Has the syntax: `value_if_true if condition else value_if_false`.

pass:

- Used as a placeholder for the body of a loop, function, or conditional statement.
- Allows you to create syntactically correct code that does nothing.

Day 2

The inner workings of functions:

- *Functions* are blocks of code that perform a specific task.
- They can take input, process it, and return output.

How to create functions:

- Use the `def` keyword followed by the function name to create a function.
- Define the parameters the function will accept within the parentheses.
- Add a block of code that will be executed when the function is called.

How to use functions that return values:

- Use the `return` statement to send a value back from the function.
- Capture the returned value in a variable when calling the function.

Day 3

Mechanism of Python Modules:

- *Modules* are files containing Python code.
- They allow for code reusability and organization.

Listing of Modules:

- Python's standard library includes a wide range of modules for various purposes.
- Third-party modules can be found on the Python Package Index (PyPI).

Importing modules from Python Standard path:

- Use the `import` keyword followed by the module name to import modules from the Python Standard Library.

Importing Modules from other Sources:

- Use the `import` keyword followed by the module name or path to import modules from other sources.

Variable in a Module:

- Variables defined in a module can be accessed by using the module name followed by the variable name.

Difference between a module and a package in Python:

- A *module* is a single file containing Python code.
- A *package* is a collection of modules in a directory with a special file named `__init__.py`.

Day 4

How to manipulate strings to a certain format:

- Use string manipulation methods like `split()`, `join()`, `strip()`, and `replace()` to modify strings.
- Utilize formatting techniques such as f-strings, `format()`, and the `%` operator for string formatting.

Grouping and capturing:

- Use parentheses `()` to group parts of a regular expression together.

- Capturing allows you to extract the content of the matched groups.

Assertions and Flags:

- *Assertions* like `^` and `$` assert the start and end of a line, respectively.

- *Flags* like `re.IGNORECASE` and `re.MULTILINE` modify how the pattern matching works in regular expressions.