Detailed Summary:

This program is a **simple finance tracker** that allows a user to:

- **Add new financial transactions** (income/expense) to a CSV file.
- **View transactions** within a specific date range, along with a **summary** (total income, total expenses, and net savings).
- **Plot the transactions** on a graph (Income vs. Expense over time).

Main components:

- **CSV Class**:
    - Manages file operations — initializes a CSV if it doesn't exist, adds new entries, and retrieves transactions between two dates.
- **add() function**:
    - Prompts the user to enter transaction details and saves them.
- **plot_transactions() function**:
    - Visualizes incomes and expenses over time using a line plot.
- **main() function**:
    - Displays a menu to either add transactions, view transactions + summary + plot, or exit the program.
- **External Imports**:
    - `pandas` for CSV manipulation,
    - `matplotlib` for plotting,
    - `datetime` for date handling,
    - and functions from a custom module `data_entry` to get user input.

Keyword	One-line Definition
`import`	Used to bring external modules or custom code into the current script.
`from … import`	Imports specific classes, functions, or variables from a module into the current script.
`pandas` (`pd`)	A Python library for data manipulation and analysis, especially for structured (tabular) data.
`csv`	A Python module for reading from and writing to CSV (Comma-Separated Values) files.
`datetime`	A module that supplies classes for manipulating dates and times.
`matplotlib.pyplot` (`plt`)	A library for creating static, animated, and interactive visualizations in Python.
`class`	A blueprint for creating user-defined objects consisting of attributes and methods.
`@classmethod`	A decorator to define a method that works with the class itself instead of instances.
`try-except` block	Structure used for handling exceptions and preventing program crashes.
`ValueError`	Exception raised when an operation receives an argument of the right type but an inappropriate value.
`raise`	Command used to trigger an exception manually in Python.
`read_csv()`	A pandas function to load a CSV file into a DataFrame.
`DataFrame`	A 2D labeled data structure in pandas, similar to a table or spreadsheet.
`to_csv()`	A pandas function to save a DataFrame as a CSV file.
`open()`	A built-in function that opens a file and returns a file object.
`DictWriter`	A csv class that writes dictionaries into a CSV file.
`writerow()`	A method that writes a single dictionary row into a CSV file.
`input()`	Built-in function to take input from the user through the console.

`float()`	A function that converts a string or number to a floating-point number.
`strptime()`	A method to convert a date string into a datetime object according to a specified format.
`strftime()`	A method to convert a datetime object into a string with a specified format.
`print()`	Built-in function to display output text to the console.
`set_index()`	A pandas method to assign a DataFrame column as the row index.
`resample()`	A pandas method to change the frequency of time-series data (e.g., daily, monthly).
`reindex()`	A pandas method to change the index (row labels) and optionally fill missing values.
`figure()`	A matplotlib method to create a new figure for plotting.
`plot()`	A method to plot data points on a graph using matplotlib.
`xlabel()`	A matplotlib function to label the x-axis of a plot.
`ylabel()`	A matplotlib function to label the y-axis of a plot.
`title()`	A matplotlib function to set the title of a plot.
`legend()`	A matplotlib function to show labels for plotted data.
`grid()`	A matplotlib function to add grid lines to a plot.
`show()`	A matplotlib function to display the plot.
`loc[]`	A pandas method for selecting data based on labels or conditional filtering.
`mask` (boolean mask)	A condition applied to a DataFrame to filter data based on True/False values.
`empty` (DataFrame property)	A property to check whether a DataFrame is empty (no rows).
`sum()`	A function that adds up numerical values.
`lower()`	A string method that converts all characters to lowercase.
`dictionary {}`	A data structure that stores key-value pairs (like `CATEGORIES`).

`recursion`	A programming technique where a function calls itself until a condition is met.
`break`	A statement used to immediately exit the nearest enclosing loop.