**Exercise: Task List with Due Dates**

Description:

You need to create a task list that allows the user to manage their tasks. Each task will have a description and a due date.

Requirements:

1. Create an empty dictionary called "task\_list" to store the tasks.

2. Implement a loop that prompts the user to enter a task description and its due date.

3. Use the `datetime` module to validate and store the due date as a `datetime` object in the dictionary.

4. Implement a loop that displays all the tasks in the task list, along with their due dates.

5. Use a set called "overdue\_tasks" to store the descriptions of tasks that are past their due dates.

6. Implement a loop that checks each task's due date. If the task is overdue, add its description to the "overdue\_tasks" set.

7. After displaying the task list, if there are any overdue tasks, display a message indicating the number of overdue tasks and list their descriptions.

Example Output:

| Task List Management System  Enter task description (or type 'exit' to finish): Finish report Enter due date (YYYY-MM-DD): 2023-06-10  Enter task description (or type 'exit' to finish): Submit presentation Enter due date (YYYY-MM-DD): 2023-05-28  Enter task description (or type 'exit' to finish): Review notes Enter due date (YYYY-MM-DD): 2023-06-02  Enter task description (or type 'exit' to finish): exit  Task List:  1. Description: Finish report, Due Date: 2023-06-10 2. Description: Submit presentation, Due Date: 2023-05-28 3. Description: Review notes, Due Date: 2023-06-02  Number of overdue tasks: 1 Overdue Tasks: - Submit presentation |
| --- |

| import datetime  task\_list = {}  while True:  print("\nTask List Management System")  description = input("Enter task description (or type 'exit' to finish): ")   if description.lower() == "exit":  break   due\_date\_str = input("Enter due date (YYYY-MM-DD): ")  due\_date = datetime.datetime.strptime(due\_date\_str, "%Y-%m-%d")   task\_list[description] = due\_date  print("\nTask List:\n") *# Write your code to generate the task list*  overdue\_tasks = set()  *# Write your code here to generate overdue tasks and store*  if overdue\_tasks:  print("\nNumber of overdue tasks:", len(overdue\_tasks))  print("Overdue Tasks:")  for task in overdue\_tasks:  print("-", task) else:  print("\nNo overdue tasks.") |
| --- |

Note: You can choose to add additional error handling and validation as per your preference.