Homework 2

You are given a dictionary `spotify` representing the top 10 most played songs on Spotify for June 2025. Each entry contains:

- A ranking (1 through 10),
- A list of artists,
- The song title,
- The length of the song in `minutes:seconds` format.

Your task is to write a Python program (.py only) that allows a user to browse this data through a menu with the following options:

Menu Options

```
Enter what you would like to browse:
    1: A list of unique artists in the top 10 most played songs
    2: Song details by ranking
    3: Songs by an artist
    4: Songs ordered by length
    0: Exit
```

Requirements

Option 1: List Unique Artists

- Display all unique artist names appearing in the top 10 songs.
- The list should be sorted alphabetically (by first letter).
- Separate artist names with a comma and space.
- Example output:

```
Alex Warren, Beson Boone, Billie Eilish, Bruno Mars, Ed Sheeran, Lady Gaga, ROSÉ,
Sabrina Carpenter
```

Option 2: Song by Ranking

- Prompt the user:
  `"Enter the ranking you're interested in (between 1 and 10): "`
- Validate user input:
  - If input is not an integer, print:
    `Invalid input. Please enter a number.`
  - If input is outside the range 1-10, print:
    `Ranking out of range.`

- If valid, print the song details in the format:
  `ranking: title by artist1, artist2, ...`
- Example:

```
# when input is 3
3: Sappire by Ed Sheeran
```

Option 3: Songs by an Artist

- Prompt the user:
  `"Enter the name of the artist you're interested in: "`
- The search should be case-insensitive.
- If one or more songs by the artist are found, print each in the format:
  `ranking: title`
- If no songs are found, print:
  `No songs were found by [artist_name]`
- Example:

```
# when input is `Lady gaga`
2: Die With A Smile
10: Abracadabra
```

Option 4: Songs Ordered by Length

- Prompt the user:
  `"Enter a number to view songs by length. (Positive: longest songs, Negative: shortest songs): "`
- Validate input as an integer. If invalid, print:
  `Invalid value. Please enter a number.`
  and return to the menu.
- Convert each song length from `"minutes:seconds"` to total seconds.
- If the number is positive, show that many songs sorted by descending length.
- If the number is negative, show that many songs sorted by ascending length.
- For each song displayed, print in the format:
  `title by artist1, artist2 (length_in_seconds seconds)`
- Examples:

```
# when input is 3
WILDFLOWER by Billie Eilish (261 seconds)
Die With A Smile by Lady Gaga, Bruno Mars (251 seconds)
Abracadabra by Lady Gaga (223 seconds)
```

```
# when input is −4
APT. by ROSÉ, Bruno Mars (169 seconds)
Espresso by Sabrina Carpenter (175 seconds)
Sappire by Ed Sheeran (179 seconds)
Beautiful Things by Beson Boone (180 seconds)
```

Option 0: Exit

- Exit the program.

Additional Notes

- Use a loop to keep displaying the menu until the user chooses to exit.
- Use `try-except` blocks to handle invalid inputs gracefully.
- Keep your outputs clear and user-friendly.
- Pay attention to proper string formatting and case insensitivity when needed.
- Do not include any packages and libraries.

Submit your complete Python script implementing the above functionality.

To test your hw2.py, install pytest and run pytest in the terminal. There are 9 test cases, and all should pass.