

 100 XP

# Exercise - Ensure that pet nicknames and personality descriptions are complete

25 minutes

The Contoso Pets app is used to help find new homes for abandoned pets. Your goal in this challenge is to develop the app features used to ensure that we have a completed dataset for each animal in the `ourAnimals` array.

## Specification

You need to develop a feature that ensures animal nicknames and personality descriptions are complete.

This feature must:

- Be enabled inside the appropriate application branch (must not overwrite the code in the code branch for menu option 2).
- Skip over any animal in the `ourAnimals` array when the value of pet ID is set to the value default value.
- Display the pet ID value and prompt the user for an updated data value if `ourAnimals` array data is missing or incomplete.
- Ensure that a valid string is assigned to `animalNickname` for all animals in the `ourAnimals` array that have assigned data.
- Ensure that a valid string is assigned to `animalPersonalityDescription` for all animals in the `ourAnimals` array that have assigned data.
- Enforce the following validation rules for `petNickname` and `petPersonalityDescription`:
  - Values cannot be null.
  - Values cannot have zero characters.
  - Any further restriction is up to the developer.

- Inform the application user when all data requirements are met, pausing the application to ensure the message can be seen and responded to.

## Check your work

To validate that your code satisfies the specified requirements, complete the following steps:

1. Use Visual Studio Code to build and run your app.
2. At the Terminal command prompt, enter **4**
3. Verify that the Terminal panel updates with a message similar to the following:

Output

```
Enter a nickname for ID #: c4
```

4. At the Terminal command prompt, press the Enter key (without typing any characters).
5. Verify that your code repeats the prompt requesting a value for the nickname of the pet.

The Terminal panel should update to display something similar to the following:

Output

```
Enter a nickname for ID #: c4
```

```
Enter a nickname for ID #: c4
```

6. At the Terminal command prompt, enter **snowflake**
7. Verify that your code accepts **snowflake** as a valid entry and that the Terminal panel displays a message similar to the following:

Output

```
Enter a personality description for ID #: c4 (likes or dislikes, tricks, energy level)
```

8. At the Terminal command prompt, press the Enter key (without typing any characters).

9. Verify that your code repeats the prompt requesting a value for the personality description of the pet.

The Terminal panel should update to display something similar to the following:

#### Output

```
Enter a personality description for ID #: c4 (likes or dislikes, tricks, energy level)
```

```
Enter a personality description for ID #: c4 (likes or dislikes, tricks, energy level)
```

10. At the Terminal command prompt, enter **loves to curl up in a warm spot**
11. Verify that your code accepts **loves to curl up in a warm spot** as a valid entry and that the Terminal panel displays a message similar to the following:

#### Output

```
Nickname and personality description fields are complete for all of our friends.  
Press the Enter key to continue
```

12. If you specified further restrictions for valid entries, run the appropriate test cases to verify your work.

Congratulations if you succeeded in this challenge!

## Next unit: Knowledge check

[Continue >](#)