Firstly, to follow the SOLID principles. Classes should have a single responsibility. These have been separated out. This includes creating a report provider class, which is responsible for creating the report. Each entity has been moved into its own class file.

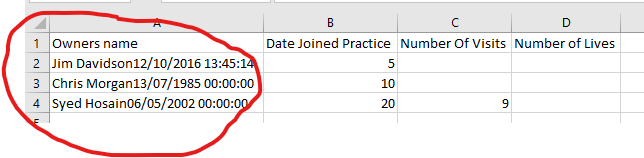
For the Open/Closed Principle, represented they the O in SOLID, I’ve moved the responsibility for creating the CSV to the relevant Pet class. This allows for more pet classes to be added in the future. Extending the application without changing the base class.

The base class (Pet) has been marked as abstract and must be inherited. It has a base method for creating the CSV string. This can be overridden by a derived class, for example in the Cat class. Allowing for polymorphism, where the method is behaving differently depending on the caller. Derived classes Cat and Dog have been marked as sealed as they should not be extended.

StringBuilder has been used to concatenated the CSV string value, this is more efficient as it doesn’t create multiple memory allocations, which would happen in the previous implementation.

Tests have been moved into a separate project, allowing for separate deployment of the main application. Tests have been given meaningful names, to explain what is being tested.

I’ve increased test coverage and found an error with the output of the CSV string. Where a comma was missing, therefore the date was in the wrong column. Please see images below. Also, time has been removed, as this is not required in this instance.



Corrected

