**Narrative: Databases**

I wanted to showcase skills in database integration, and this program could really benefit from such a feature. This would allow saving records in persistent storage, instead of temporary memory that is lost when program execution terminates. To start, I created a database for dogs and a database for monkeys, using MapDB configured as a hash map. I used the animal’s name as the key and the animal object as the value. Then, any time the user adds an animal, the animal is added it to the temporary ma and also to the persistent storage. The Dog, Monkey, and RescueAnimal classes required implementing Serializable and adding a serialVersionUID to each, in order to work properly. At initial program execution, the records in permanent storage are loaded into the in-program data structures to ensure fast access. Finally, I was sure to close each database, only if the database is not empty. Now, the animal objects persist across program executions, and valuable information is not lost. Hooray!

Throughout this enhancement, I have met the following course outcomes:

* + - * + Design and evaluate computing solutions that solve a given problem using algorithmic principles and computer science practices and standards appropriate to its solution, while managing the trade-offs involved in design choices
        + Demonstrate an ability to use well-founded and innovative techniques, skills, and tools in computing practices for the purpose of implementing computer solutions that deliver value and accomplish industry-specific goals

I have also met the other three course outcomes as follows:

* + Employ strategies for building collaborative environments that enable diverse audiences to support organizational decision making in the field of computer science
  + Design, develop, and deliver professional-quality oral, written, and visual communications that are coherent, technically sound, and appropriately adapted to specific audiences and contexts
  + Develop a security mindset that anticipates adversarial exploits in software architecture and designs to expose potential vulnerabilities, mitigate design flaws, and ensure privacy and enhanced security of data and resources