

ROBOT VS. DINOSAUR PROJECT FRAMEWORK

Introduction

In previous projects, you became adjusted to the practice of creating the flow of your programs by writing functions and calling functions – this is an approach known as the “functional programming” paradigm.

Now, with Object Oriented Programming, we are taking a different approach: creating classes to represent the unique data structures in our program, creating instances (objects) from those classes, and having those objects interact with each other in order to create the flow of our program. This creates a “Russian Nesting Doll” effect of all your classes nested inside of larger classes that represent the larger units of functionality for the program.

Be sure to read the User Stories document thoroughly to get a clear idea of what features need to be present for this project, and follow the steps below to get started. Good luck!

 *MAKE SURE YOU HAVE THE UML FOR THIS PROJECT BEFORE*

STARTING!

Resources

Lectures

- Classes & Objects
- UML (Unified Modeling Language)

Documents

- UML – Robot vs. Dinosaur

Relevant Projects

- Classes & Objects Lab

Other Resources

- **Code Demo – Building Classes from a UML**

Tasks

1. Review all necessary materials for getting started
2. Create your project (including setting up your GitHub repository for source control)
3. Create all of the classes for the project based off of the UML
4. Starting with the smallest class, write the methods for your classes.
5. Work through the user stories as you write methods, testing each method before moving on!

Setup Steps

1. Create a folder for your project, then create a GitHub repository for the project.
2. Clone down the repository to your computer and put the invisible `.git` folder inside your project folder (as well as the `.gitignore` and `README`). Make an initial commit.

3. Create a new file for each class on the UML, as well as a `main.py` file that will serve as the entry point of your application.
4. Begin working on the user stories by filling in your classes from smallest to large. Begin with the `Weapon` class, then move on to the `Dinosaur` class, then the `Robot` class.
5. Finally, fill out the methods for your battle logic in the `Battlefield` class. You will only need to import the `Dinosaur` and `Robot` classes into the `Battlefield` class.
6. You will run the game by creating an object from the `Battlefield` class inside of `main.py` and calling the `run_game` method!

End Result

Please see the Project Walkthrough – Robot vs. Dinosaur video for an idea of how the final product will look!