* testDiffTowerCostDifficulty
  + M3 Test by Evan Leleux
  + Asserts that the cost of each tower is different for each difficulty
  + Tests Easy vs Medium, Easy vs Hard, and Medium vs Hard
* testBuyingTowerWithInsuccifientMoney
  + M3 Test by Evan Leleux
  + Asserts that the game will not allow a tower to be bought with insufficient money
  + Tests Tower 1,2,3
* testHealthDifferentForDifficulties
  + M3 Test by Andrew Fitton
  + Asserts that the player’s health is different for each different difficulty
  + Tests Easy vs Medium, Easy vs Hard, Medium vs Hard
* testMoneyDifferentForDifficulties
  + M3 Test by Andrew Fitton
  + Asserts that the player’s money is different for each different difficulty
  + Tests Easy vs Medium, Easy vs Hard, Medium vs Hard
* testUpdateMoneyWhenTowerBought
  + M3 Test by Sonya Yuen
  + Asserts that the player’s money is updated when a tower is bought
  + Tests Easy, Medium, and Hard levels with buying different towers
* testCorrectStartingMoney
  + M3 Test by Sonya Yuen
  + Asserts the correct money for each level
* testCorrectStartingHealth
  + M3 Test by Erich Drawdy
  + Asserts the correct starting health for each level
* testCostIntervals
  + M3 Test by Erich Drawdy
  + Asserts that tower cost doubles from easy to hard level, and that tower cost increases 1.5x from easy to medium levels.
  + Tests Easy vs Medium, Easy vs Hard
  + Tests Towers 1, 2, 3
* testCorrectHealthPercentage
  + M3 Test by Braden Dunaway
  + Asserts simple decimals to set health values divided by the total health
  + Tests a variety of percentages in different difficulties
* testDifficultyToString
  + M3 Test by Braden Dunaway
  + Asserts the expected string for a given difficulty level
  + Tests unset diff, easy, medium, hard, and levels above hard
* testConsistentEnemyValues
  + M4 Test by Evan Leleux
  + Ensures that initialized enemy objects retain their correct values after being cast to their parent type enemy class
* testEnemyHandlerImages
  + M4 Test by Evan Leleux
  + Ensures that images in the EnemyHandler currentEnemies are different instances from the default ones in possibleEnemies.
  + Makes sure that each enemy is unique and has their own stats.
* testDifferentEnemySpeeds
  + M4 Test by Andrew Fitton
  + Asserts that different enemy types have different speeds
  + Tests Panther vs Eagle, Panther vs Blue Devil, Eagle vs Blue Devil
* testDifferentEnemyHealth
  + M4 Test by Andrew Fitton
  + Asserts that different enemy types have different healths
  + Tests Panther vs Eagle, Panther vs Blue Devil, Eagle vs Blue Devil
* testDifferentEnemyRewards
  + M4 Test by Sonya Yuen
  + Asserts that different enemies have different reward amounts
  + Tests Blue Devil vs Panther, Panther vs Eagle, Eagle vs Bulldog, Bulldog vs Bluedevil
* testDifferentEnemyStrength
  + M4 Test by Sonya Yuen
  + Asserts that different enemies have the correct strength amounts
  + Blue devil: 10 strength, Panther: 4 strength, Eagle: 1 health, Bulldog: 30 health
* testTakeDamage
  + M4 Test by Erich Drawdy
  + Tests to make sure take damage works, and that isAlive switches to false after enemy health is less than or equal to 0.
* testDifferentWeight
  + M4 Test by Erich Drawdy
  + Tests to make sure enemies have different weights
* testForEnemyVisibility
  + M5 Test by Andrew Fitton
  + Asserts that enemies are visible until they take damage to 0 health
  + Tests for each enemy type
* testForDamageToMonument
  + M4 Test by Andrew Fitton
  + Asserts that enemies do damage to the player’s health
  + Tests for each enemy type
* testCorrectTowerNames
  + M5 Test by Sonya Yuen
  + Asserts that the correct towers have the correct corresponding name
* testGainMoneyAfterDeath
  + M5 Test by Erich Drawdy
  + Asserts that the player gains money after an enemy dies.
* testCorrectSpeed
  + M5 Test by Erich Drawdy
  + Asserts that the speed is correct for each enemy.
* testTowerUpgradeCost()
  + M5 Test by Braden Dunaway
  + Asserts that the upgrade cost increases with differing levels/costs
* testUpgradeBoostsStats()
  + M5 Test by Braden Dunaway
  + Asserts that the strength and radius are changing by the correct amount for each tower upgrade
* testIncreaseKillCount()
  + M6 Test by Evan Leleux
  + Asserts that kill count increases after an enemy death
* testShootOnce()
  + M6 Test by Evan Leleux
  + Asserts that a tower correctly sets the shot once Boolean after attacking any enemy
* testBossHealth ()
  + M6 Test by Andrew Fitton
  + Asserts that the boss has a health greater than any other enemies
* testBossStrength()
  + M6 Test by Andrew Fitton
  + Asserts that the boss has a strength greater than any other enemies
* testIncreaseMoneySpent()
  + M6 Test by Erich Drawdy
  + Tests to ensure that money spent counter increases when money is spent
* testBossReward()
  + M6 Test by Erich Drawdy
  + Tests to check that the boss has a reward that is greater than other enemies
* testBossImage()
  + M6 Test by Sonya Yuen
  + Asserts that the boss is visually different than other enemies
* testUpgradeBoughtCount()
  + M6 Test by Sonya Yuen
  + Asserts that upgrades bought count increases when a tower is upgraded
* testEnemyAtMonument()
  + M6 Test by Braden Dunaway
  + Asserts the switching of the atMonument Boolean for a given Enemy
* testEnemySlow()
  + M6 Test by Braden Dunaway
  + Asserts that the speed of an enemy and the current speed are not equal
  + Uses a modified attackEnemy() method to test the slow effect