

Lab 2: Inheritance

Instruction

1. Click the provided link on CourseVille to create your own repository.
2. Open Eclipse and then “File > new > Java Project” and set project name in this format **2110215_Lab2_2019_1_{ID}_{FIRSTNAME}**
 - Example: **2110215_Lab2_2019_1_6131234521_Alice**.
3. Initialize git in your project directory
 - Add .gitignore.
 - Commit and push initial codes to your GitHub repository.
4. Implement all the classes and methods following the details given in the problem statement file which you can download from CourseVille.
 - You should create commits with meaningful messages when you finish each part of your program.
 - Don't wait until you finish all features to create a commit.
5. Test your codes with the provided JUnit test cases, they are inside package **test.grader**
 - If you want to create your own test cases, please put them inside package **test.student**
 - Aside from passing all test cases, your program must be able to run properly without any runtime errors.
6. After finishing the program, create a UML diagram using **ObjectAid** and put the result image (**UML.png**) at the root of your project folder.
7. Export your project into a jar file called **Lab2_2019_1_{ID}** and place it at the root directory of your project.
 - Example: **Lab2_2019_1_6131234521.jar**
8. Push all other commits to your GitHub repository

1. Problem Statement: CPArena

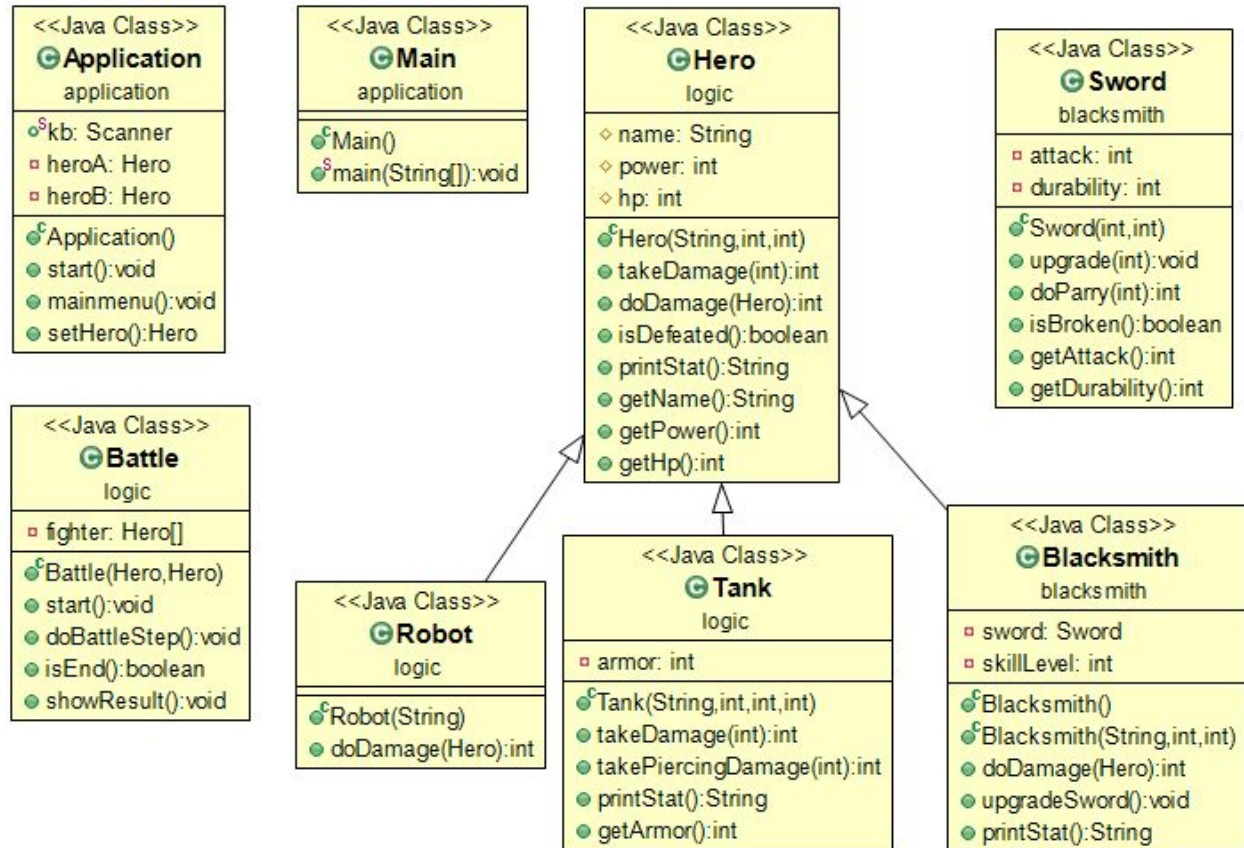
Because battle royale video games are too mainstream, CP decided to create a game to not let them to take over the console market. This game is called “CPArena”.

```
Welcome to CPArena
=====
[1] Change Hero A
[2] Change Hero B
[3] Start The Battle
Please select your option:
```

CPArena is a game where the player will create **Heroes** to beat each other until someone is knocked out in a gladiator style. Some hero are not regular heroes. Some are **Tank** with an armor that can reduce incoming damage. Some are mass-produced **Robot** which all have the same stat and ignore any armor while attacking. Some are **Blacksmith** who wield a **Sword** which can be upgraded every time after an attack. Heroes' HP is never regenerated after fight as the gladiators never give up and leave the arena. Let's the CPArena BEGIN!

2. Implementation Details:

The diagram of the program is illustrated below:



You can **ignore** classes in **application package** as you don't need to do anything to complete this assignment. In order to complete the other classes, you need to understand OOP, JUnit, and Inheritance. **Any class that are not in the initial files but appear in this diagram must be created by yourself.**

* Noted that Access Modifier Notations can be listed below

- + (public)
- # (protected)
- (private)

2.1 Class: Sword

This class is inside the package `blacksmith` with JUnit Test Case: `SwordTest`. Class `Sword` might be seen as completed but this class is not working correctly. Your task is using JUnit to create test cases for methods this class in `SwordTest` which we partially provided and use them to find and fix the mistakes which **might have more than one**.

2.1.1 Fields

- int attack	Sword's attack value. This show how much damage that sword wielder will do to target.
- int durability	Sword's durability value. This show how long the sword can withstand.

2.1.2 Constructors

+ Sword(int attack, int durability)	<p>Initialize a Sword with the according attack and durability value.</p> <p>Noted: Sword attack should not be less than 1 when created. If so, consider it as 1. It is also the same for Sword durability.</p>
-------------------------------------	--

2.1.3 Methods

+ void upgrade(int val)	This upgrade Sword attack by val and increase its durability by 1. If val is negative, only just increase its durability.
+ int doParry(int attack)	<p>The Sword is used to parry an attack. If the attack is a nonpositive value, consider it as 1.</p> <p>After parried, decrease this Sword durability by 1 and reduce incoming attack power by its attack value.</p> <p>This function return a value of incoming damage after reduction but it never be less than 0. If it happened, return incoming damage as 0.</p>
- boolean isBroken()	The Sword will be considered as broken if its durability is no longer more than 0. Return true or false based on is it broken or not.

2.2 Class Hero

This class represent a Hero in CPArena. **You have to create this inside package logic.**

2.2.1 Fields

# String name	Hero's name.
# int power	Hero's power, how much damage this Hero can do to the other.
# int hp	Hero's hp, how much damage can this Hero withstand.

2.2.2 Constructors

+ Hero(String name, int power, int hp)	Initialize a Hero with the following parameter. However, minimum Hero power is 0 and minimum Hero hp is 1.
--	--

2.2.3 Methods

+ int takeDamage(int damage)	<p>The Hero is taking damage, reduced their hp by that value. If damage is negative, they receive 0 damage. Hero's hp after receive damage shall never go below 0.</p> <p>This method return damage value they takes.</p>
+ int doDamage(Hero target)	<p>The Hero is attacking to the target. The target take the Hero's power as damage.</p> <p>This method return damage value dealt to the target.</p>
+ boolean isDefeated()	<p>Return if this Hero is defeated. A Hero is defeated if its hp is reach 0 or below.</p>
+ String printStat()	<p>Return String with the following format : "[atk:%power% hp:%hp%]"</p> <p>Example: "[atk:9000 hp:9000]"</p>
Getter methods for each field	

2.3 Class Tank extends Hero

Tank is Hero subclass. They do mostly identical to parent class with a few extra details, every Tank has an armor to reduce incoming attack damage. **You have to create this inside package logic.**

2.3.1 Fields

- int armor	Tank's armor reduces incoming damage by its value.
-------------	---

2.3.2 Constructors

+ Tank(String name, int power, int hp, int armor)	Initialize Tank with the following parameter. Every Tank must have at least 1 armor.
---	---

2.3.3 Methods

+ int takeDamage(int damage)	This method override its parent Tank's armor reduces incoming damage by its value. The method return damage value that Tank receives. However, incoming damage can't be reduced to be less than 0. Tank's hp shall never go below 0, otherwise keep it at 0.
+ int takePiercingDamage(int damage)	However, some Heroes can attacking Tank ignore its armor. Tank take piercing damage without reduction. Tank's hp shall never go below 0, otherwise keep it at 0. This method return damage value that Tank receives.
+ String printStat()	Return String with the following format : " [atk:%attack% hp:%hp% armor:%armor%] " Example: "[atk:9000 hp:9000 armor:9000]"
+ getArmor()	Return Tank's armor value.

2.4 Class Robot extends Hero

Robot is mass-produced Hero class with advance weapon such as Tesla Coil which can ignore target armor. Every robot has the same stat and attack differently from the other class. It has no additional field. **You have to create this inside package logic.**

2.4.1 Constructors

+ Robot(String name)	Initialize a Robot with the 2 power and 10 hp . Its name is according to the parameter.
----------------------	--

2.4.2 Methods

+int doDamage(Hero target)	This method override its parent A Robot is so advanced in technology that it's attack can penetrate Tank's armor, dealing piercing damage to Tank by its power. It still do normal damage to other Hero class. This method return damage value dealt to the target.
----------------------------	--

2.5 Class Battle (partially provided)

This class represents a battle in CPArena. We already partially provide code but still need you to complete it. This class is inside package logic.

2.5.1 Fields

- Hero[] fighter	This array contain Heroes in the battle. Each battle in CPArena is 1 vs 1.
------------------	--

2.5.2 Constructors

+ Battle(Hero a, Hero b)	Initialize Battle with the following fighters.
--------------------------	--

2.5.3 Methods

+ void start()	Start a Battle with the following steps: <ol style="list-style-type: none">1. Print "==== Here Come A New Challenger ====="2. Print name and stat from each fighter3. Start doBattleStep loop until the Battle is end4. Show result
+ void doBattleStep()	In this step, each fighter attack each other simultaneously. Report attacker's damage done to target and target's hp after an attack. Also show if anyone is defeated yet or not.
+ boolean isEnd()	Return if Battle is ended or not. A Battle ended when any fighter is defeated.
+ void showResult()	Show the battle result. There are 3 possible outcome: <ul style="list-style-type: none">- First fighter win- Second fighter win- Draw

2.6 Class Blacksmith ***OPTIONAL CHALLENGE***

This give no extra credit nor count toward assignment credit. You can either skip this or honing your skill by implementing this class. If you want to complete this class, uncomment some specific comments in class **Application** to enable this class to be called. **This class shall be created inside package blacksmith.**

2.6.1 Fields

- Sword sword	Blacksmith always wield a Sword which can be upgraded every time after attacking
- int skillLevel	Blacksmith's skillLevel represent how much they can upgrade their Sword after an attack.

2.6.2 Constructors

+ Blacksmith()	Initialize default Blacksmith name: "DefaultBlacksmith" power: 1 hp: 1 skillLevel : 1 sword: 1,1
+ Blacksmith(String name, int hp, int skillLevel)	Initialize Blacksmith with the following parameter and equipped with 1,1 Sword. Because Blacksmith depend on their Sword more than their strength, Blacksmith power is always 1.

2.6.3 Methods

+ int doDamage(Hero target)	This method override its parent Blacksmith do damage base only on their Sword's attack value. After an attack, the Sword is upgraded once.
+ void upgradeSword()	Upgrade the Sword by Blacksmith's skillLevel.
+ String printStat()	Return a String with format: "[hp:%hp% lvl:%skillLevel%]" Example: [hp:9000 lvl:9000]

3. Test Scenario

Welcome to CPArena

=====

[1] Change Hero A

[2] Change Hero B

[3] Start The Battle

Please select your option: 3

=====

==== Here Come A New Challenger ====

== DefaultHero A [atk:1 hp:1] VS DefaultHero B [atk:1 hp:1] ==

DefaultHero A do 1 damage to DefaultHero B

DefaultHero B has 0 HP left

DefaultHero B do 1 damage to DefaultHero A

DefaultHero A has 0 HP left

DefaultHero A is knocked out

DefaultHero B is knocked out

== DRAW ==

Returning to main menu

=====

[1] Change Hero A

[2] Change Hero B

[3] Start The Battle

Please select your option: 1

=====

Please enter new hero name: DIO

[1] Normal

[2] Tank

[3] Robot

Please enter new hero type: 1

Please enter new hero power and hp: 100 9000

== New Hero created: DIO [atk:100 hp:9000] ==

Returning to main menu

=====

[1] Change Hero A

[2] Change Hero B

[3] Start The Battle

Please select your option: 2

=====

Please enter new hero name: JoJo

[1] Normal

[2] Tank

[3] Robot

Please enter new hero type: 2

Please enter new hero power, hp, and armor: 50 500 10

== New Tank created: JoJo [atk:50 hp:500 armor:10] ==

Returning to main menu

=====

[1] Change Hero A

[2] Change Hero B

[3] Start The Battle

Please select your option: 3

=====

==== Here Come A New Challenger ====

== DIO [atk:100 hp:9000] VS JoJo [atk:50 hp:500 armor:10] ==

DIO do 90 damage to JoJo
JoJo has 410 HP left

JoJo do 50 damage to DIO
DIO has 8950 HP left

DIO do 90 damage to JoJo
JoJo has 320 HP left

JoJo do 50 damage to DIO
DIO has 8900 HP left

DIO do 90 damage to JoJo
JoJo has 230 HP left

JoJo do 50 damage to DIO
DIO has 8850 HP left

DIO do 90 damage to JoJo
JoJo has 140 HP left

JoJo do 50 damage to DIO
DIO has 8800 HP left

DIO do 90 damage to JoJo
JoJo has 50 HP left

JoJo do 50 damage to DIO

DIO has 8750 HP left

DIO do 90 damage to JoJo

JoJo has 0 HP left

JoJo do 50 damage to DIO

DIO has 8700 HP left

JoJo is knocked out

== DIO WIN ==

Returning to mainmenu

=====

[1] Change Hero A

[2] Change Hero B

[3] Start The Battle

Please select your option: 1

=====

Please enter new hero name: T-800

[1] Normal

[2] Tank

[3] Robot

[4] Blacksmith

Please enter new hero type: 3

== New Robot created: T-800 [atk:2 hp:10] ==

Returning to main menu

=====

[1] Change Hero A

[2] Change Hero B

[3] Start The Battle

Please select your option: 2

=====

Please enter new hero name: Will Smith

[1] Normal

[2] Tank

[3] Robot

[4] Blacksmith

Please enter new hero type: 4

Please enter new hero hp and skill level: 10 1

== New Blacksmith created: Will Smith [hp:10 lvl:1] ==

Returning to main menu

=====

[1] Change Hero A

[2] Change Hero B

[3] Start The Battle

Please select your option: 3

=====

==== Here Come A New Challenger ====

== T-800 [atk:2 hp:10] VS Will Smith [hp:10 lvl:1] ==

T-800 do 2 damage to Will Smith

Will Smith has 8 HP left

Will Smith do 1 damage to T-800

T-800 has 9 HP left

T-800 do 2 damage to Will Smith

Will Smith has 6 HP left

Will Smith do 2 damage to T-800

T-800 has 7 HP left

T-800 do 2 damage to Will Smith

Will Smith has 4 HP left

Will Smith do 3 damage to T-800

T-800 has 4 HP left

T-800 do 2 damage to Will Smith

Will Smith has 2 HP left

Will Smith do 4 damage to T-800

T-800 has 0 HP left

T-800 is knocked out

== Will Smith WIN ==

Returning to main menu