ACS-1904 W2024
Lab #6
Due by Friday, February 16 at 11:59 pm

- Compress all of the .java files in this project into a .zip file called Lab6.zip.
- Submit the .zip file containing the Java classes for this project to the Nexus drop box
- **Include your name and student number in each file as a comment.**

To complete Lab 6 you will need a complete and correct version of the Ships hierarchy from Lab 5. You can get my version here https://github.com/rsveinson/ACS-1904-2024-Lab5Solutions.git, or you can use yours if you're confident that it is up to par.

Make the following changes:

NOTE: the Enum stuff is optional.

1) Add an Enum to the project called Propulsion. (**Optional**)
   a) Propulsion has 3 Enum constants.
      i) DIESEL
      ii) DIESELELECTRIC
      iii) GASTURBINE
   b) Include the following fields in the Enum with the indicated values, both are String.

`displayName`	`model`
Diesel	RT96-Turbo
Diesel Electric	Codag-DE Hybrid
Gas Turbine	GE-LM2500

   c) No getters
   d) Override the toString() in Propulsion so that it prints the display name and the model in the following format. You can also see the sample output for clarification.

```
Diesel: RT96-Turbo
```

2) Changes to `Ship.java`
   a) Make Ship an abstract class.
   b) Change the name of the `name` field in Ship to `shipName`
   c) Add a getter and setter for `shipName`.
   d) Replace the current `getName()` method with an abstract method of the same name.

Steps e-h are optional:

   e) Add a new field to the Ship class of type Propulsion, that's the Enum that you created in step 1.
   f) Change the toString() method in Ship so that it includes the toString of the Enum. It should return a String in this format. Also, see the sample output for clarification.

```
Ranger: 1997
Diesel: RT96-Turbo
```

   g) The full-arg constructor should now look something like this: note the last parameter.

```
public Ship(String n, String f, int y, Propulsion p){
```

h) You will have to add a line in the full-arg constructor to assign the parameter p to the new propulsion field. Something like: `propulsion = p;` where propulsion is the name of the field and p is the parameter.

3) Changes to `Tanker.java` and `CruiseShip.java` **(Optional: only if you are adding the Enum)**

a) The only changes needed in the `Tanker` and `CruiseShip` classes are to update the parameter list in the full-arg constructors (see 4a for a hint) and update the argument list in the call to the superclass constructor.

4) Changes to `Ships.java`

a) Change the instantiations to include the Enum and change the Ship to a Tanker:

If you are not adding the Enum don't include the Propulsion argument.

```
Tanker s = new Tanker("Evergreen", "Panama", Propulsion.DIESELELECTRIC, 2004, 300000,
500);
Tanker t = new Tanker("Ranger", "Beliz", Propulsion.DIESEL, 1997, 250000, 450);
CruiseShip c = new CruiseShip("Aurora", "HongKong", Propulsion.GASTURBINE, 2021,
"MSV", 5453);
```

b) Make the necessary changes to the driver code so that the output matches the sample output below. (**Optional: only if you are adding the Enum)**

Sample output:

```
Evergreen, 300000L, 500m
Ranger, 250000L, 450m
MSV, Aurora, 5453

Printing with toString() and the new Enum information
Ranger: 1997
Diesel: RT96-Turbo

 end of program
```

Submit your Lab6.zip file that includes all lab files (Ship.java, Tanker.java, CruiseShip.java, Ships.java, and Propulsion.java) via Nexus. Note Propulsion.java is included only if you are completing the optional part of the lab. Also Note: only files with a .zip extension will be accepted by the drop box in Nexus.