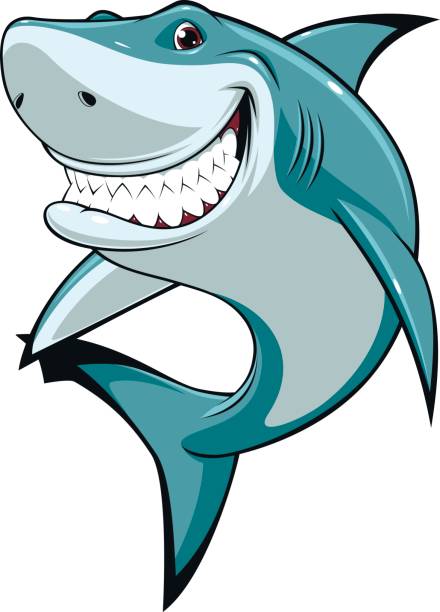
# 03. Shark Taxonomy



*Let's dive into the amazing world of the ocean, where we'll meet some incredible creatures called sharks. Sharks come in many different types, like hammerheads and reef sharks, and they each have their own unique features. In the 'Sharks Taxonomy' challenge, we'll create a list to keep track of these cool underwater animals and learn more about them.*

## Preparation

Download the provided skeleton in Judge. **Do not** change the **StartUp** class or its **namespace**.

**Ensure that the project is named SharkTaxonomy, and all classes, fields, and methods should be named exactly as presented in the document. Maintain the project structure as described below.**

## Problem description

Your task is to **create a repository that stores sharks** by implementing the following classes.

### Shark

First, create a class **Shark** with the following properties:

* **Kind - string**
* **Length - int**
* **Food - string**
* **Habitat - string**

The class **constructor** should receive **kind, length, food and habitat**.

Override the **ToString()** method in the following format:  
**"{Kind} shark: {Length}m long.  
Could be spotted in the {Habitat}, typical menu: {Food}"**

### Classifier

**Next**, create a class **Classifier** that has **Species** (a collection for storing sharks). All entities inside the repository have the **same properties**. The **Classifier** class should have the following **properties**:

* **Capacity - int**
* **Species - List<Shark>**
* **GetCount** – getter => **returns** the **number** of all **sharks.**

The class **constructor** should receive **capacity.** Also, it should initialise the **Species** with a **new instance** of the collection.

#### Implement the following features:

* **Method AddShark(Shark shark)** – **adds** a **Shark** to the **Species** **collection**, **if** the **Capacity** **allows it**. If there is a **Shark** from the **same Kind already added**, do not duplicate sharks, just **skip the command.**
* **Method RemoveShark(string kind)** – attempts to find a **Shark**, within the **Species** collection,with **Kind value, matching the given parameter**. If no **Shark is found**, the method returns **false. Otherwise,** it is **removed from the collection** and the method returns **true**.
* **Method GetLargestShark()**– **returns the ToString() value** ofthe **largest of all sharks, arranged** by **length.**
* **Method GetAverageLength() – returns** the average length of **the sharks added to the collection.**
* **Method Report()** – **returns** a **string** in the following **format**:
  + **"{count} sharks classified:  
    {shark1}  
    {shark2}  
    {…}**

**{sharkn}"**

## Constraints

* You will always have **sharks added** **before** receiving commands **manipulating** the collections in   
  the **Classifier**.
* There will always be a **Shark** that **has the greatest value** of **Length** property.

## Examples

This is an example of how the **Classifier** class is **intended to be used**.

|  |
| --- |
| **Sample code usage** |
| //Initialize new repository (Classifier)  Classifier classifier = new(10);  //Initialize entities (Shark)  Shark greatWhite = new("Great White", 5, "Seals", "Open Ocean");  Shark hammerhead = new("Hammerhead", 4, "Fish", "Tropical Waters");  Shark tiger = new("Tiger", 4, "Turtles", "Coral Reefs");  Shark mako = new("Mako", 3, "Fish", "Open Ocean");  Shark bull = new("Bull", 3, "Fish", "Rivers");  Shark whale = new("Whale", 12, "Plankton", "Open Ocean");  Shark leopard = new("Leopard", 1, "Crabs", "Shallow Waters");  Shark goblin = new("Goblin", 4, "Deep-sea Creatures", "Deep Ocean");  Shark thresher = new("Thresher", 6, "Fish", "Open Ocean");  Shark blacktipReef = new("Blacktip Reef", 2, "Fish", "Coral Reefs");  Shark oceanicWhitetip = new("Oceanic Whitetip", 3, "Fish", "Open Ocean");  //Add sharks to the repository  classifier.AddShark(greatWhite);  classifier.AddShark(hammerhead);  classifier.AddShark(tiger);  classifier.AddShark(mako);  classifier.AddShark(bull);  classifier.AddShark(whale);  classifier.AddShark(leopard);  classifier.AddShark(goblin);  classifier.AddShark(thresher);  classifier.AddShark(blacktipReef);  //Check collection count  Console.WriteLine(classifier.GetCount);  //10  //Attempt to add a shark that will exceed the capacity of the Classifier  classifier.AddShark(oceanicWhitetip);  //Check collection count  Console.WriteLine(classifier.GetCount);  //10  //Remove existing shark  classifier.RemoveShark("Blacktip Reef"); //Returns True  //Check collection count  Console.WriteLine(classifier.GetCount);  //9  //Try to remove not existing shark  classifier.RemoveShark("Blue"); //Returns False  //Check collection count  Console.WriteLine(classifier.GetCount);  //9  //Try to add once again a shark, if there is enough capacity already  classifier.AddShark(oceanicWhitetip);  //Check collection count  Console.WriteLine(classifier.GetCount);  //10  //Get the shark that has the greatest body length  Console.WriteLine(classifier.GetLargestShark());  //Whale shark: 12m long.  //Could be spotted in the Open Ocean, typical menu: Plankton  //Get the average Length of all sharks added to the collection  Console.WriteLine(classifier.GetAverageLength());  //4.5  //Print Sharks Report  Console.WriteLine(classifier.Report());  //10 sharks classified:  //Great White shark: 5m long.  //Could be spotted in the Open Ocean, typical menu: Seals  //Hammerhead shark: 4m long.  //Could be spotted in the Tropical Waters, typical menu: Fish  //Tiger shark: 4m long.  //Could be spotted in the Coral Reefs, typical menu: Turtles  //Mako shark: 3m long.  //Could be spotted in the Open Ocean, typical menu: Fish  //Bull shark: 3m long.  //Could be spotted in the Rivers, typical menu: Fish  //Whale shark: 12m long.  //Could be spotted in the Open Ocean, typical menu: Plankton  //Leopard shark: 1m long.  //Could be spotted in the Shallow Waters, typical menu: Crabs  //Goblin shark: 4m long.  //Could be spotted in the Deep Ocean, typical menu: Deep - sea Creatures  //Thresher shark: 6m long.  //Could be spotted in the Open Ocean, typical menu: Fish  //Oceanic Whitetip shark: 3m long.  //Could be spotted in the Open Ocean, typical menu: Fish |

## Submission

Zip all the files in the project folder except **bin** and **obj** folders.