Address class

// Properties \\

 **Street Name** – String (e.g., "Main Street")

 **House/Building Number** – String (Some addresses include letters, e.g., "12B")

 **City** – String (e.g., "New York")

 **State/Province/Region** – String (e.g., "California" or "Ontario")

 **Postal/ZIP Code** – String (e.g., "10001" or "W1A 1AA")

 **Country** – String (Ensure it's validated using an API)

Country class

// Properties \\ // Use API for checking if it’s valid?

1. Name – String (Setter Use API to check if country is real)
2. Diving regulations – WHAT TYPE?
3. Languages – Linked List just for fun!

Person class

// Properties \\

1. ID – String ( validate ID from function I made )
2. First Name – String ( REGEX)
3. Last Name – String ( REGEX)
4. Date of Birth – String (Validate before saving)

Diver class : Person

// Properties \\

1. Unique diver ID – String ( Make something up )
2. ID – String ( : person)
3. First Name – String ( : person)
4. Last Name – String ( : person)
5. Date of Birth – String ( : person)
6. Diving log => getting services from diving clubs (needs to be kept well can be destroyed easly)

Diver Instructor class : Diver

// Diving Instructors will approve and document the dives they are doing

// Properties \\

1. Start working – Date
2. End working – Date
3. Current Working Club - DivingClub Object
4. Clubs Worked[] – DivingClub Object array

Diving Club Class

// Properties \\

1. License – String ( REGEX ) (MUST BE UNIQUE)
2. static string[] Licenses = [] // Will store all of the licenses that were made
3. Name – String ( REGEX)
4. Contact name – String
5. Address – Address Object ( Will include in it Country – OBJECT)
6. Phone number - String ( REGEX)
7. email – String ( REGEX )
8. hasSite – Bool
9. WEBsite address ( REGEX )
10. Link to diving regulations ( line 15 in WORD GUIDE ) – Will be included
    1. At least 2 divers? BOOL
    2. Certification Requirement
    3. Depth Limitations
    4. Environmental Protection? BOOL
11. DivingLog – DivingInfo[]

Diving Site Class

// Properties \\

1. name – string
2. Address – Address Object ( Will include in it Country – OBJECT)
3. description – string (small desc on the site)
4. Length – double (meters)
5. Width – double (meters)
6. Deep – double (meters)
7. Water Salty – bool.

Signature Class

// Properties \\

Diver – Diver Object (The diver that signed)

DateTime – DateTime

Signature – FIND A WAY TO REPRESENT THE SIGNATURE

Diving Info Class

// Properties \\

1. Diving Club – Diving Club Object
2. Diving site – Diving Site Object
3. Dive date – date
4. HeadDiver = diver instruction class (The one who will fill the information)
5. HeadDiver Works currently at club? BOOL (must be true)
6. Diver Instructors[] – Array of diver instruction class
7. Divers[] - Array of diver class
8. Water entre – date time
9. Water exit – date time
10. Divers Signutures[] – Array of diver signutures
11. Water temperature – double
12. Tide level double (0-1) 1 = high, 0 = low, in the middle will be in between, will be shown in %
13. Was dive valid – BOOL (will be set after all data input based on)
    1. diving regulations – ALL VALID?
    2. Head diver works currently?
    3. All divers signed?

**Information:**

* Diving is usually in pairs
* Hard to find a partner
* Not all needed equipment is always available

Diving **MUST**

* Diving journey will happen only when in a diving club (else show error!)
  + Need ability to add diving clubs!!
* At the end of the dive, all divers must sign that they have been diving

**SYSTEM SCREENS**

To log in to system

// Dictionary that will contain all of those details

Username == Email.

Password == ( REGEX: 8 digits exactly [a-z0-9] ONLY!)

Find a diving club

// Finding a club will be used by one of the following (Select box with a few options that will contain)

1. Club name
2. Club License
3. Club Contry
4. Club Address