What are APIs, Libraries, and Packages?

This article explains what APIs, libraries, and packages are as well as describing the differences between them and how they are used.

You may have heard the terms API, Library, and Package and have wondered what the differences between them are. It is true that they are closely related, but it can be confusing when trying to describe what each individual part is responsible for when coding.

Let's start with APIs!

The acronym "API" is short for Application Program Interface and it is what allows applications to interact with another piece of software. In Java terms, it describes the list of all libraries and packages which are available to use in your code.

An API is really a description of how to interact with the software and the format to use when interacting with it. You can find a detained API specification for the Java libraries and packages here: Java 8 API The API documentation is extremely important for understanding how objects of classes will behave as well as the attributes contained within them. It also allows for abstraction to take place which means that the underlying code from software accessed from an API is not shown.

An easy way to think about APIs in terms of software libraries is that it contains information about the libraries, packages, and classes which you can use. The API does not contain the actual code to implement classes, but instead contains the specifications of those classes.

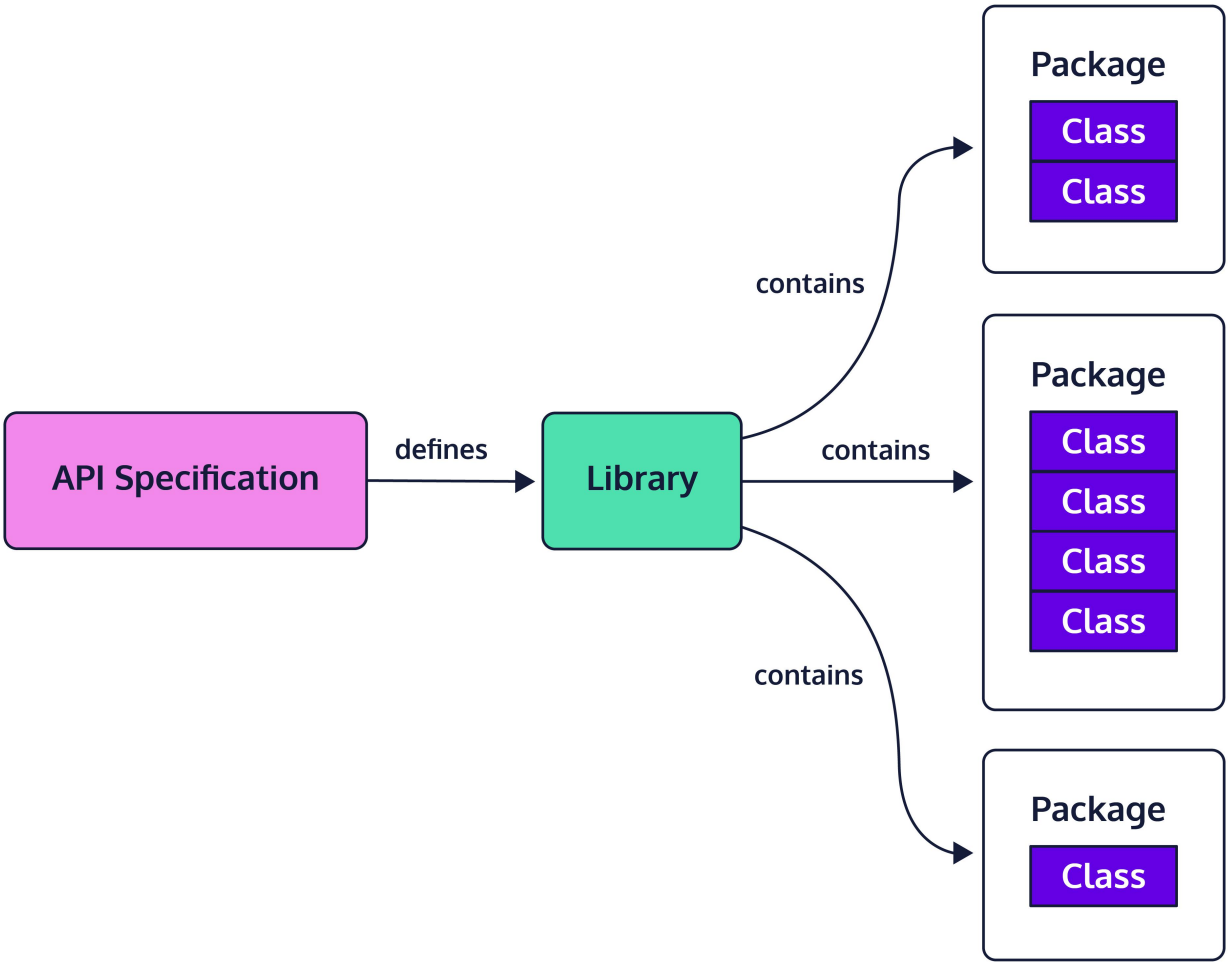
So what is a library?

We have learned that APIs describe the functionality and interactions with a piece of software, but they do not contain any code. This is where libraries come in. Libraries refer to the collection of code that implements the API. This can consist of multiple packages that contain the code covering the classes specified in the API.

Then what is a package?

A package is a group of related classes and interfaces in Java. They organize related files together and allow easy access to classes that provide a certain type of functionality. For example, we can think of the `java.util` package. This package contains many useful Java utility classes such as the `Random`, `Array`, and `ArrayList` classes. Another example would be the `java.io` package. The classes contained within that package would all be related to input and output functionality such as the `File`, `Reader`,

and `Writer` classes. The classes in the library (which cover the API specifications) are organized and grouped into packages.



This image shows the relationship between each of the components.

Let's test what you know about APIs, Libraries, and Packages so far!

Multiple Choice Question

1. What does an API do?

   A. An API let's us know the details about classes and packages in a library. It is made up of the code which defines those classes and packages.
   B. An API let's us know the details about classes and packages in a library. It is not the code itself.
   C. An API is the code for classes and packages defined in a library.
   D. An API does not contain any real code or information about classes and packages.

2. What is the purpose of a library?

    A. The library is a location for the code which implements the API specifications. This contains all packages and classes.
    B. The library is located within a package and contains the code for a specified class.
    C. In the context of APIs, a library is a location where you can rent books.
    D. The library is a location to store the specifications for desired classes and packages.

3. What is a package used for?

    A. A package contains all classes and interfaces which implement the API specifications.
    B. Packages are groups of API specifications used for defining classes.
    C. Packages are the files containing java code which provide some functionality.
    D. A package contains a group of classes and interfaces which are related in some way.

How can we use APIs, Libraries, and Packages?

All of these parts are being used whenever you import some code in your project. For example, when you want to create an `ArrayList` object in your code, you will import the `ArrayList` class from the `java.utils` package located within the Java Class Library (JCL). The `java.utils` package implements the functionality for the `ArrayList` class as described in the Java API Specification.

Each part works together to allow you to import outside code to use in your projects. From these imported classes, you can create objects from them, call static methods from them, and use them as if they were any other class that you created. Because of this, the Java API and libraries simplify complex programming tasks because you can re-use code that has been written before to handle similar problems. Imagine having to re-create the logic required to access your operating system and retrieve a file from a directory manually every time you want to load a text file in your project. Instead of doing all that work yourself, you can use the JCL to use the provided File class.

Speaking of which, let's try an example using the `File` class!

Code Challenge

There is a hidden message located in a file called `message.txt`. Use the `File` and `Scanner` classes to print out the contents of the text file to the console. Use the Java API located at Java API to look up how to use the Java classes. Some of the

code has been provided for you. To complete the code, you need to import and initialize the `Scanner` object.

```
import java.io.File;
import java.io.IOException;

public class MessageReader {
  public static void main(String[] args) throws
IOException {
    File file = new File("./message.txt");
    //TODO: Import and initialize a scanner
object then uncomment the while loop below to
print the message.

    // while(scan.hasNext()) {
    //     System.out.println(scan.nextLine());
    // }
  }
}
```

Nice work! Our Java toolbelt is ever-expanding! We can now begin to use APIs, Libraries, and Packages built into Java!

Hint

You'll want to import `java.util.Scanner;`. Then create a new `Scanner` using `Scanner scan = new Scanner(file);`