

# Programming Assignment 02

Emre Caniklioglu  
21803577

December 2021

## 1 Parallel File Downloader

Parallel File Downloader console app's source code contains a main method and a class with the name 'ParallelFileDownloader' this class within itself contains three private static classes. First one is called 'RequestWriter', second one is called 'RequestGet' and third one is called 'GetRequestASYNC'.

## 2 Request Writer

This class contains three static methods. All of them create a 'PrintWriter' object and implement their suggested HTTP protocol by writing to the 'PrintWriter'. Their names are: 'GetRequestWriter', 'GetHeadRequestWriter', 'GetRangeRequestWriter'.

## 3 Request Get

This class contains static methods for connecting the Request Writer's static methods to the java web socket api. It contains three static functions which connect the name suggested RequestWriter to the java sockets API. Their names are 'SendGet', 'SendGetHead', 'Send-GetRange'. These functions take a hostname, path, lower and upper bound if needed. And they return a List<String>, where each element is a line from the response. Only the GET head functions result list contains the response header.

## 4 GetRequestASYNC

This class contains a static function with the name 'SendRequest' which sends a HTTP GET range request to the given host and path between the lower bound and upper bound in another thread. And returns the Future object of that thread.

## 5 Parallel File Downloader

This class's constructor takes a URL and the number of threads to use it then calls the 'InitializeQueue' function which tries to send a get head request to the index file and parse its results. If there is an exception it displays the error message and quits. If it is successful

for each path in the index page it sends a head request and if its response is OK than it divides the file size and sends range request to the URL using the 'GetRequestASYNC' class and collects the Future's in a list. It waits for all of the futures to finish. Afterwards it iterates over the list while combining the result to a new list. If any error occurs it displays an error message and exits. If there are no errors it proceeds by saving the file to the local computer and continuing to the next path.