LeetCode Questions:

Q1. Two Sum

public class Solution {

public int[] TwoSum(int[] nums, int target) {

Dictionary<int, int> map = new Dictionary<int, int>();

int[] result = new int[2];

for (int i = 0; i < nums.Length; i++) {

int complement = target - nums[i];

if (map.ContainsKey(complement)) {

result[0] = map[complement];

result[1] = i;

return result;

}

map[nums[i]] = i;

}

return result;

}

}

Q9. Palindrome Number

public class Solution {

public bool IsPalindrome(int x) {

string xs = x.ToString();

for (int i = 0; i<xs.Length/2; i++){

if(xs[i] != xs[xs.Length-1-i])

{

return false;

}

}

return true;

}

}

Q217. Contains Duplicates—Solution 1

public class Solution {

public bool ContainsDuplicate(int[] nums) {

HashSet<int> seen = new HashSet<int>();

for(int i = 0;i<nums.Length; i++){

if(seen.Contains(nums[i])){

return true;

}

seen.Add(nums[i]);

}

return false;

}

}

OR

Q217. Contains Duplicates—Solution 2

Q412. Fizz Buzz

public class Solution {

public IList<string> FizzBuzz(int n) {

List<string> k = new();

for (int i = 1; i <= n; i++)

{

if (i % 3 == 0 && i % 5 == 0) {

k.Add("FizzBuzz");

} else if (i % 3 == 0) {

k.Add("Fizz");

} else if (i % 5 == 0) {

k.Add("Buzz");

} else {

k.Add($"{i}");

}

}

return k;

}

}