**CODING CHALLENGES:**

1. Write a Java program to implement Queue Using Array And Class

public class QueueUsingArrayMain {

private int capacity;

int queueArr[];

int front;

int rear;

int currentSize = 0;

public QueueUsingArrayMain(int sizeOfQueue) {

this.capacity = sizeOfQueue;

front = 0;

rear = -1;

queueArr = new int[this.capacity];

}

public void enqueue(int data) {

if (isFull()) {

System.out.println("Queue is full!! Can not add more elements");

} else {

rear++;

if (rear == capacity - 1) {

rear = 0;

}

queueArr[rear] = data;

currentSize++;

System.out.println(data + " added to the queue");

}

}

public void dequeue() {

if (isEmpty()) {

System.out.println("Queue is empty!! Can not dequeue element");

} else {

front++;

if (front == capacity - 1) {

System.out.println(queueArr[front - 1] + " removed from the queue");

front = 0;

} else {

System.out.println(queueArr[front - 1] + " removed from the queue");

}

currentSize--;

}

}

public boolean isFull() {

if (currentSize == capacity) {

return true;

}

return false;

}

public boolean isEmpty() {

if (currentSize == 0) {

return true;

}

return false;

}

public static void main(String a[]) {

QueueUsingArrayMain queue = new QueueUsingArrayMain(5);

queue.enqueue(6);

queue.dequeue();

queue.enqueue(3);

queue.enqueue(99);

queue.enqueue(56);

queue.dequeue();

queue.enqueue(43);

queue.dequeue();

queue.enqueue(89);

queue.enqueue(77);

queue.dequeue();

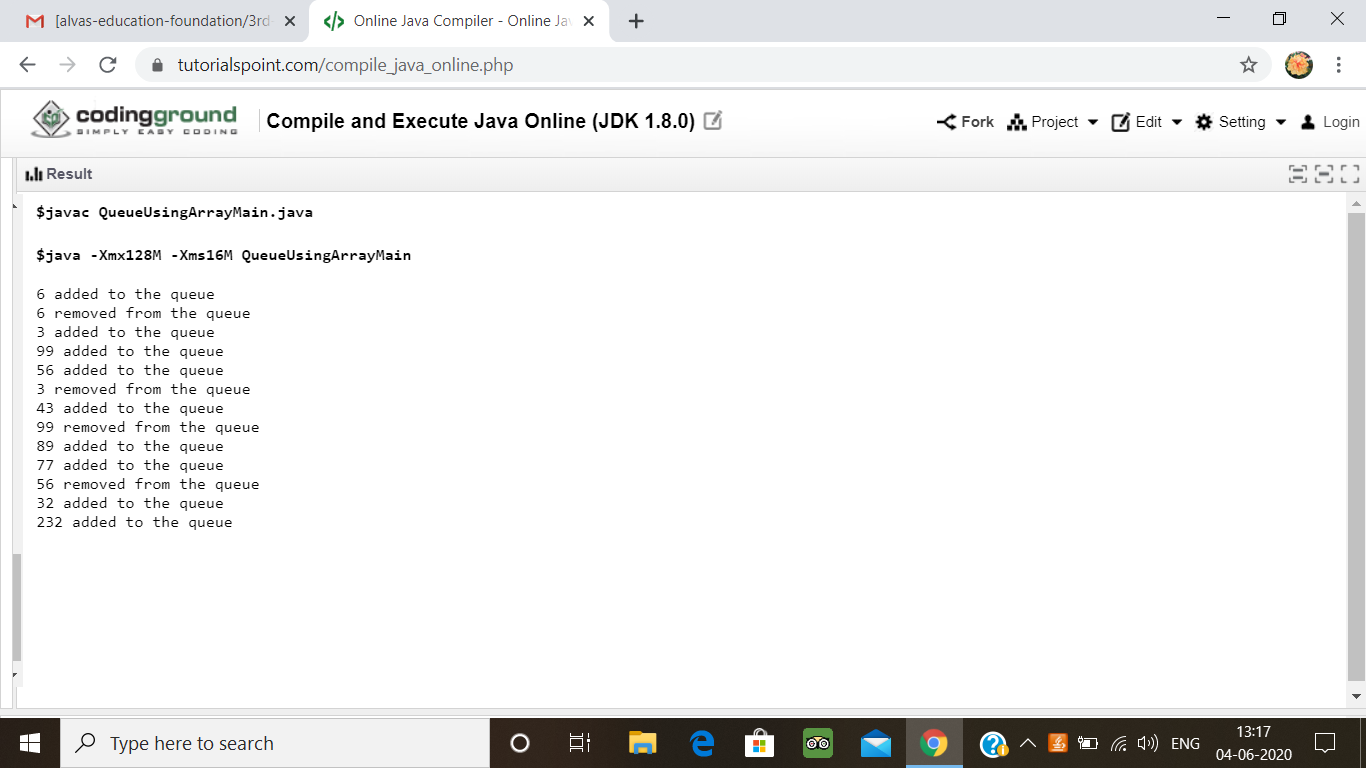
queue.enqueue(32);

queue.enqueue(232);

}

}

**Output:**



1.Python program to combine the strings

Description:

Take two strings, return a string of the form short+long+short, with the shorter string on the outsides and the longer string on the inside. The strings will not be the same length, but they may be empty (length 0).

Eg:  
Str1= 'Hello'  
Str2= 'Hi'  
Combined\_str('Hello', 'Hi') ='HiHelloHi'

str1=input('enter first string: ')

str2=input('enter second string:')

a=len(str1)

b=len(str2)

if(a<b):

print(str1+str2+str1)

else:

print(str2+str1+str2)

**output:**

****