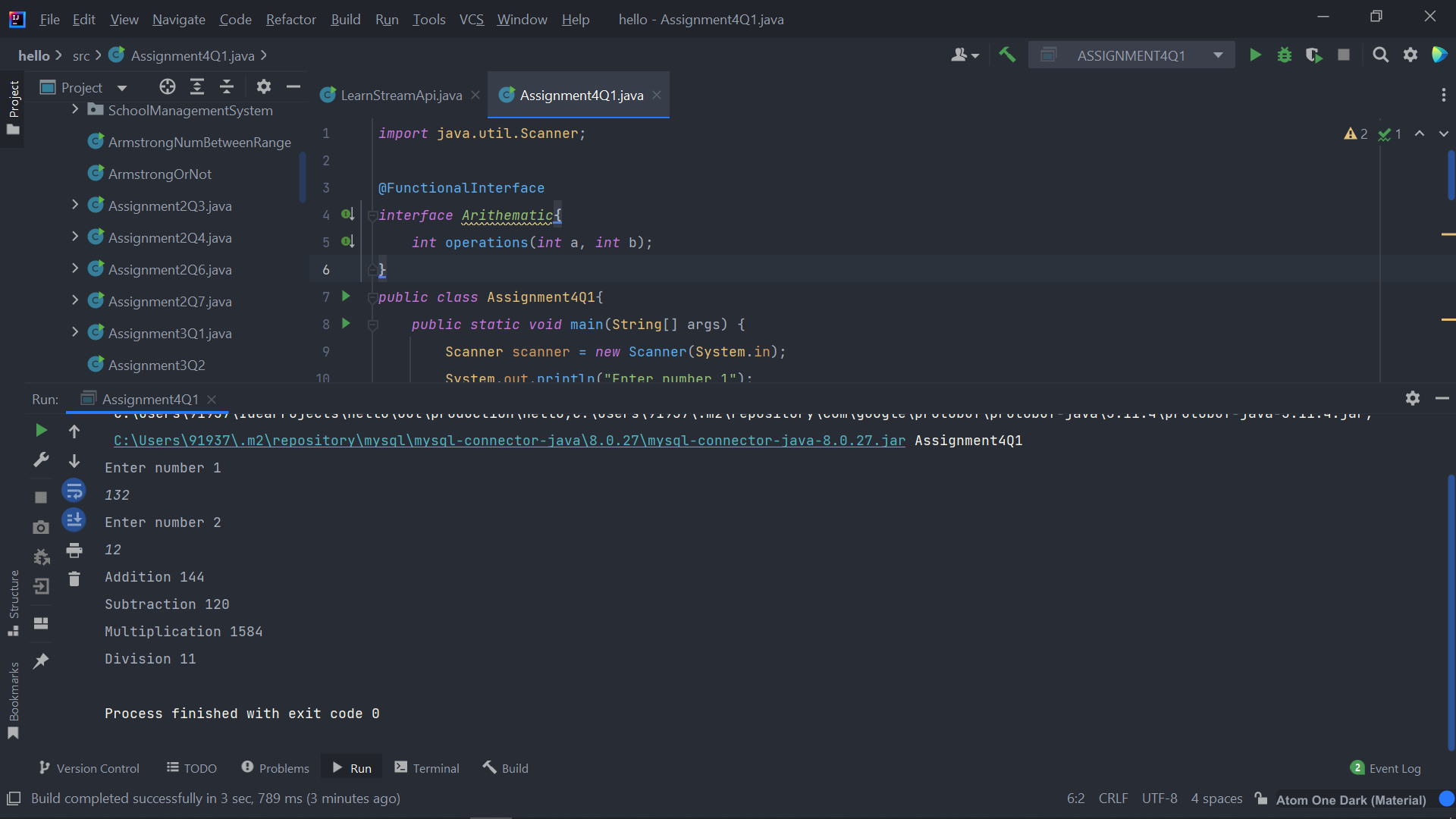
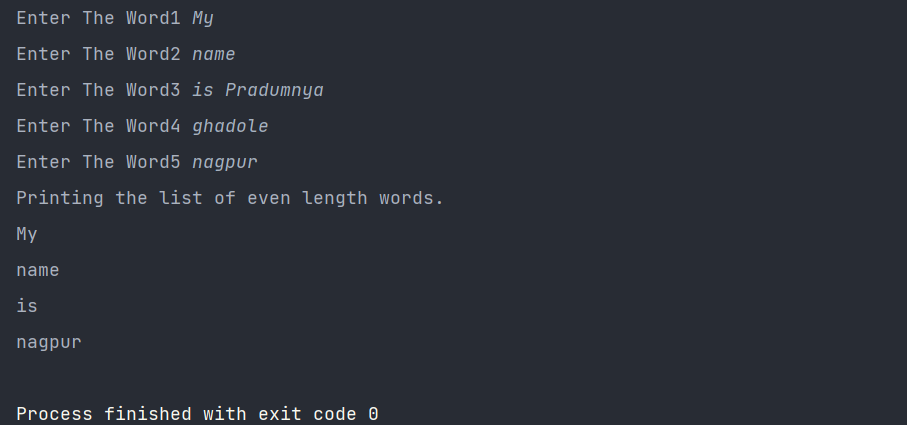
Q1. *import* java.util.Scanner;  
  
@FunctionalInterface  
*interface Arithematic*{  
 *int* operations(*int* a, *int* b);  
}  
*public class* Assignment4Q1{  
 *public static void* main(String[] args) {  
 Scanner scanner = *new* Scanner(System.in);  
 System.out.println("Enter number 1");  
 *int* a1 = scanner.nextInt();  
 Scanner scanner1 = *new* Scanner(System.in);  
 System.out.println("Enter number 2");  
 *int* b1 = scanner1.nextInt();  
  
 *Arithematic* addition = (a, b) -> (a + b);  
 System.out.println("Addition "+ addition.operations(a1,b1));  
  
 *Arithematic* subtraction = (a, b) -> (a - b);  
 System.out.println("Subtraction "+subtraction.operations(a1,b1));  
  
 *Arithematic* multiplication = (a, b) -> (a \* b);  
 System.out.println("Multiplication "+multiplication.operations(a1,b1));  
  
 *try* {  
 *Arithematic* division = (a, b) -> (a / b);  
 System.out.println("Division "+ division.operations(a1, b1));  
 }  
 *catch* (ArithmeticException e){  
 System.out.println(e);  
 }  
  
 }  
}



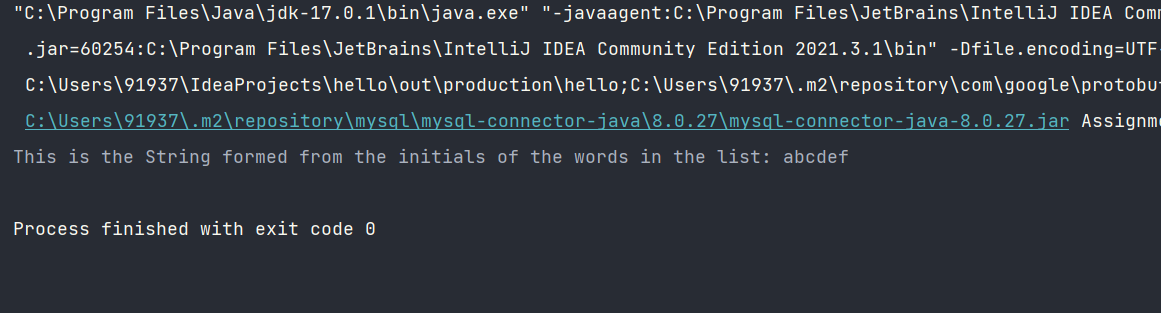
Q2. *interface Check*{  
 *void* check(*int* totalPrice, String status);  
}  
  
*public class* Assignment4Q2 {  
 *public static void* main(String[] args) {  
  
 *Check* check = (totalPrice, status)->{  
 *if* ((totalPrice > 10000) && (status == "Accepted" || status == "Completed")){  
 System.out.println(totalPrice+" "+ status);  
 }  
 };  
 check.check(12090,"Completed");  
 check.check(43590,"Completed");  
 check.check(1290,"Accepted");  
 check.check(13450,"Started");  
 check.check(4330,"Completed");  
 check.check(125690,"Started");  
 }  
}



Q4. *import* java.util.ArrayList;  
*import* java.util.Scanner;  
  
*public class* Assignment4Q4 {  
  
 *public static void* main(String[] args) {  
 ArrayList<String> arrayList = *new* ArrayList<>();  
  
  
 *for* (*int* i = 1; i <=5; i++) {  
 Scanner scanner = *new* Scanner(System.in);  
 System.out.print("Enter The Word"+i+" ");  
 arrayList.add(scanner.next());  
 }  
  
 arrayList.removeIf(n -> n.length()%2 != 0);  
  
 System.out.println("Printing the list of even length words. ");  
 *for* (String s:arrayList) {  
 System.out.println(s);  
 }  
 }  
}

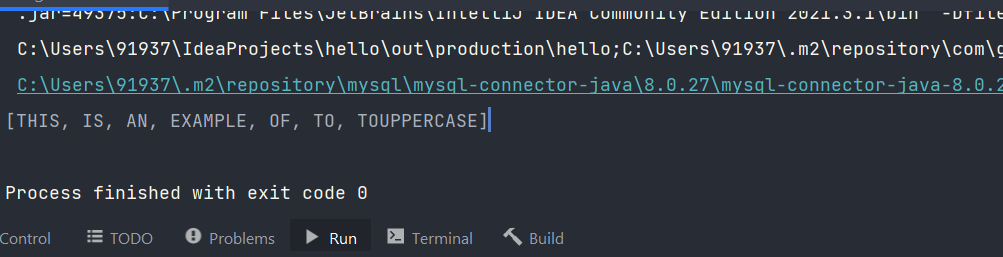


Q5. *import* java.util.Arrays;  
*import* java.util.*List*;  
  
*public class* Assignment4Q5 {  
 *static List*<String> *list* = Arrays.*asList*("alpha", "bravo", "charlie", "delta", "echo", "foxtrot");  
  
 *interface ProcessWords*{  
 String processWords(*List*<String> list);  
 }  
  
 *public static ProcessWords* processWords(){  
 *return*(list) ->{  
 String result = "";  
 *for* (String i:list) {  
 result += i.charAt(0);  
 }  
 *return* result;  
 };  
 }  
 *public static void* main(String[] args) {  
 String answer;  
 answer = *processWords*().processWords(*list*);  
 System.out.println("This is the String formed from the initials of the words in the list: "+answer);  
 }  
}



Q6.

*import* java.util.Arrays;  
*import* java.util.*List*;  
  
*public class* Assignment4Q6 {  
 *public List*<String> convertToUpperCase(*List*<String> list) {  
 list.replaceAll(String::toUpperCase);  
 *return* list;  
 }  
 *public static void* main(String[] args) {  
 *List*<String> stringList = Arrays.*asList*("This","is","an","example","of","to","toUppercase");  
 Assignment4Q6 assignment4Q6 = *new* Assignment4Q6();  
 System.out.println(assignment4Q6.convertToUpperCase(stringList));  
 }  
}



Q7.

*import* java.util.HashMap;  
*import* java.util.*Map*;  
*import* java.util.stream.Collectors;  
  
*/\*\*  
 \* entrySet() gets an iterable sequence of entries  
 \* stream() creates a stream for that iterable  
 \* map() converts that stream of entries into a stream of strings of the form "key - value"  
 \* collect(Collectors.joining(", ")) joins all the entries in the stream into a single string,  
 \* using ", " as the separator. Collectors.joining is a method which returns a Collector which  
 \* can work on an input sequence of strings, giving a result of a single string.  
 \*/  
public class* Assignment4Q7 {  
 *public static void* main(String[] args) {  
 *Map*<String,Integer> map = *new* HashMap<>();  
 map.put("Food",1);  
 map.put("Good",2);  
 map.put("mood",3);  
 map.put("Stood",4);  
 Assignment4Q7 assignment4Q7 = *new* Assignment4Q7();  
 System.out.println(assignment4Q7.convertKeyValueToString((HashMap<String, Integer>) map));  
 }  
 *public* String convertKeyValueToString(HashMap<String, Integer> map) {  
 String result = map.entrySet().stream()  
 .map(entry -> entry.getKey() + "-" + entry.getValue())  
 .collect(Collectors.*joining*(","));  
 *return* result;  
 }  
}



Q8.

*import* java.util.ArrayList;  
*import* java.util.*List*;  
  
*public class* Assignment4Q8 {  
 *public static void* main(String[] args) {  
 *List*<Integer> list = *new* ArrayList<>();  
 list.add(12);  
 list.add(32);  
 list.add(45);  
 list.add(2);  
 list.add(7);  
  
 *new* Thread(() ->{  
 *for* (Integer i: list) {  
 System.out.println(i);  
 }  
 }).start();  
 }  
}

