**Practical-1**

**Switch Case**

**Code:**

import 'dart:io';

void main() {

  print("Enter your day: ");

  String? input = stdin.readLineSync();

  String day = "";

  if (input != null) day = input.toLowerCase();

  switch (day) {

    case "monday":

      print("Today is $day");

      break;

    case "tuesday":

      print("Today is $day");

      break;

    case "wednesday":

      print("Today is $day");

      break;

    case "thursday":

      print("Today is $day");

      break;

    case "friday":

      print("Today is $day");

      break;

    case "saturday":

      print("Today is $day");

      break;

    case "sunday":

      print("Today is $day");

      break;

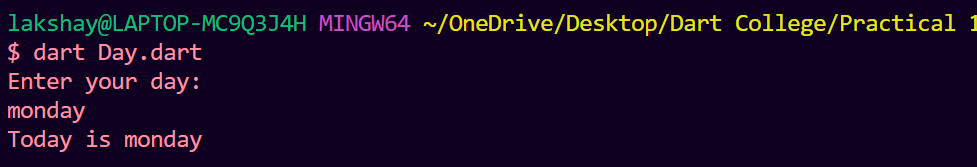
    default:

      print("Enter valid day!");

  }

}

**Output:**

****

**If Else**

**Code:**

import 'dart:io';

void main() {

  print("Enter your grade: ");

  String? input = stdin.readLineSync();

  int? marks = int.parse("$input");

  if (marks >= 0 && marks <= 100) {

    if (marks >= 90)

      print("Grade: A+");

    else if (marks < 90 && marks >= 80)

      print("Grade: B+");

    else if (marks < 80 && marks >= 70)

      print("Grade: C+");

    else if (marks < 70 && marks >= 60)

      print("Grade: D+");

    else if (marks < 60 && marks >= 40)

      print("Grade: E+");

    else

      print("Your have FAILED!");

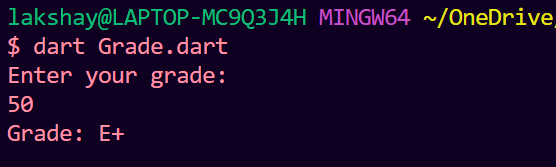
  } else {

    print("Invalid input! \nEnter marks between 0 to 100!");

  }

}

**Output:**

****

**Factorial**

**Code:**

import 'dart:io';

void main() {

  int ans = 1;

  print("Enter your number: ");

  String? fact = stdin.readLineSync();

  try {

    int? factorial = int.parse("$fact");

    for (int i = 1; i <= factorial; i++) {

      ans \*= i;

    }

    print("Factorial is: $ans");

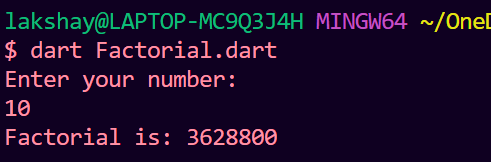
  } catch (e) {

    print("Enter valid number!");

  }

}

**Output:**

****

**Fibonacci**

**Code:**

import 'dart:io';

void main() {

  print("Enter fibonacci limit: ");

  String? input = stdin.readLineSync();

  int? limit = int.parse("$input");

  int first = 0, second = 1;

  for (int i = 2; i < limit; i++) {

    int third = second + first;

    first = second;

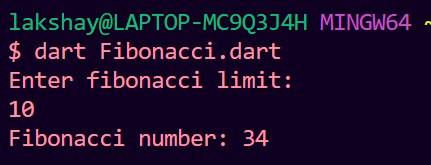
    second = third;

  }

  print("Fibonacci number: $second");

}

**Output:**

****

**Prime:**

**Code:**

import 'dart:io';

void main() {

  print("Enter Number: ");

  String? input = stdin.readLineSync();

  try {

    int? num = int.parse("$input");

    int flag = 0;

    if (num <= 0 || num == 1)

      flag = 1;

    else {

      for (int i = 2; i \* i < num; i++) {

        if (num % i == 0) {

          flag = 1;

          break;

        }

      }

    }

    flag == 1 ? print("$num is not a Prime!") : print("$num is a Prime!");

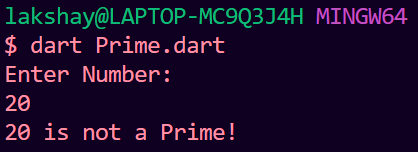
  } catch (e) {

    print("Enter valid umber!");

  }

}

**Output:**

****

**Practical-2**

**Guess Number**

import 'dart:io';

import 'dart:math';

void main() {

  Random random = new Random();

  int myRandom = random.nextInt(100);

  while (1 == 1) {

    print("Guess your number: ");

    String? input = stdin.readLineSync();

    int num = 0;

    try {

      num = int.parse("$input");

    } catch (e) {

      print("Please Enter a valid number!");

      continue;

    }

    if (num == myRandom) {

      print("Correct Guess! \nNumber is: $num");

      break;

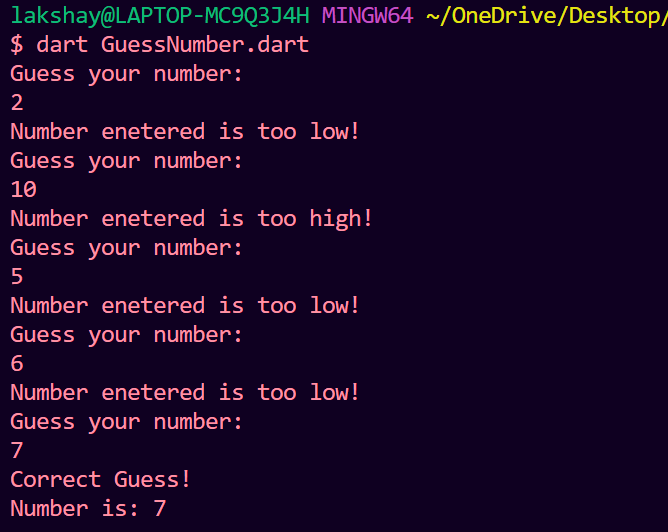
    } else if (num > myRandom)

      print("Number enetered is too high!");

    else if (num < myRandom) print("Number enetered is too low!");

  }

}

****

**Palindrome**

import 'dart:io';

void main() {

  print("Enter your number/ string: ");

  String? input = stdin.readLineSync();

  String? reversed = input!.split('').reversed.join('');

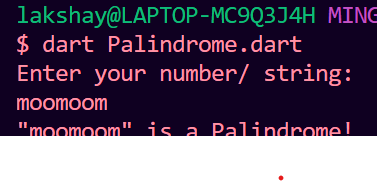
  if (reversed == input)

    print("\"$input\" is a Palindrome!");

  else

    print("\"$input\" is not a Palindrome!");

}

****

**Remove Duplicates**

void main() {

  var input = ["lakshay", "geet", "lakshay", "ritesh", "Disha", "Disha"];

  var ans = [];

  for (int i = 0; i < input.length; i++) {

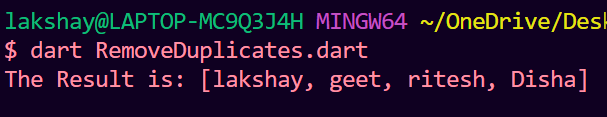
    if (ans.contains(input[i])) continue;

    ans.add(input[i]);

  }

  print("The Result is: $ans");

}

****

**Reverse Element List and Array Operations**

void main() {

*// 1.Create a list variable and store 6 three digit integer values in it. Create another list which contains reverse elements of this list.*

  var input = <int>[111, 222, 333, 444, 555, 666];

  var ans = [];

  for (int i = input.length - 1; i >= 0; i--) {

    ans.add(input[i]);

  }

  print("Reversed input list: $ans");

*// 2. Use the following List functions in the above list:*

*//  i.First*

*//  ii.  isEmpty*

*//  iii.  isNotEmpty*

*//  iv.  Length*

*//  v.  Last*

*//  vi.  Reverse*

*//  vii.  Single*

*//  viii.  add()    //   list.add(<element>);*

*//  ix.  addAll()     //list.addAll([<element list separated by comma>]);*

*//  x.  insert()     //list.insert(<index>,<value>);*

*//  xi.  insertAll()   //list\_names.insertAll(<index>, [<list\_of\_value>]);*

*//  xii.  replaceRange()    //list\_names.replaceRange(int start\_val, int end\_val,*

*//  xiii.  remove()   //list\_names.remove(value)*

*//  xiv.  removeAll()   //list\_name.removeAt(int index)*

*//  xv. removeLast()  //list\_names.removeLast()*

*//  xvi. removeRange()   //list\_names. removeRange(int start, int end);*

*// i*

  print(ans.first);

*// ii*

  print(ans.isEmpty);

*// iii*

  print(ans.isNotEmpty);

*// iv*

  print(ans.length);

*// v*

  print(ans.last);

*// vi*

  print(ans.reversed);

*// vii*

*// print(ans.single);*

*// viii*

  ans.add(123456);

*// ix*

  ans.addAll(<int>[-1, -2, -3]);

*// x*

  ans.insert(0, -69);

*// xi*

  ans.insertAll(1, [-9999, -8888]);

*// xii*

  ans.replaceRange(2, 5, []);

*// xiii*

  ans.remove(666);

*// xiv not given*

*// xv*

  ans.removeLast();

*// xvi*

  ans.removeRange(3, 5);

*// 3. Cerate the following set variables and perform the following operations:*

  var a = <int>{10, 11, 12, 13, 14, 15};

  var b = <int>{12, 18, 29, 43};

  var c = <int>{2, 5, 10, 11, 32};

*// a. Union of a, b and c*

  print(a.union(b.union(c)));

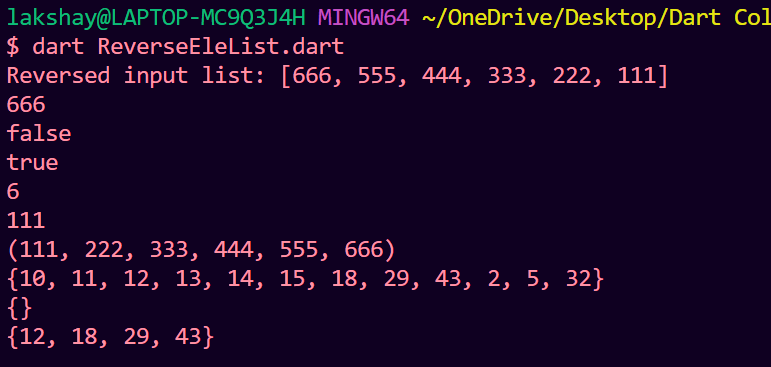
*// b. Intersection of a and b*

  print(a.intersection(b.intersection(c)));

*// c. Difference of b and c*

  print(b.difference(c));

}

****

**Reverse String**

import 'dart:io';

void main() {

  print("Enter your String: ");

  String? input = stdin.readLineSync();

  input = input == null ? "" : input;

  String ans = "", word = "";

  input.runes.forEach((int rune) {

    var character = new String.fromCharCode(rune);

    if (character == ' ') {

      ans = word + " " + ans;

      word = "";

    } else

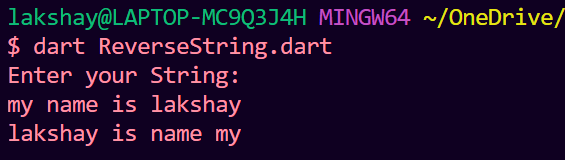
      word = word + character;

  });

  ans = word + " " + ans;

  print(ans);

}

****

**Practical-3**

**Bank**

import 'dart:math';

class Bank {

  String? name;

  int? accNum;

  int? balance;

  String? type;

  Bank(String? name, [int? accNum, int? balance, String? type]) {

*this*.name = name;

*this*.accNum = accNum ?? Random().nextInt(1000000) + 1000;

*this*.balance = balance ?? 0;

*this*.type = type ?? "user";

  }

  void deposit(int depositAmount) {

*this*.balance = *this*.balance ?? 0 + depositAmount;

  }

  int withdraw(int withdrawAmount) {

    if ((*this*.balance ?? 0) >= withdrawAmount) {

*this*.balance = *this*.balance ?? 0 - withdrawAmount;

      return withdrawAmount;

    }

    return -1;

  }

  void display() {

    print("$accNum belongs to $name of type: $type with balance: $balance");

  }

}

import 'dart:io';

import 'Bank.dart';

void main() {

  print("Enter name:");

  String? name = stdin.readLineSync();

  print("Enter Account Number:");

  String? s = stdin.readLineSync();

  int? acc = int.parse("$s");

  print("Enter balance:");

  s = stdin.readLineSync();

  int? bal = int.parse("$s");

  print("Account Type?:");

  String? type = stdin.readLineSync();

  Bank bank = new Bank(name, acc, bal, type);

  bool choice = true;

  while (true) {

    print(

        "Press 1 to withdraw money\nPress 2 to deposit money\nType -1 to exit.");

    s = stdin.readLineSync();

    int? c;

    try {

      c = int.parse("$s");

    } catch (e) {

      print("\n PLEASE ENTER VALID CHOICE: \n");

      continue;

    }

    if (c == 1) {

      print("Enter withdraw amount:");

      s = stdin.readLineSync();

      int ammount = int.parse("$s");

      int withdraw = bank.withdraw(ammount);

      if (withdraw == -1)

        print("Insufficient balance");

      else {

        int? balance = bank.balance;

        print("Withdrawed: $withdraw\nBalance: $balance");

      }

    } else if (c == 2) {

      print("Enter deposit amount:");

      s = stdin.readLineSync();

      int ammount = int.parse("$s");

      bank.deposit(ammount);

      print("Deposited $ammount Successfully!");

    } else if (c == -1) {

      break;

    } else {

      print("\n PLEASE ENTER VALID CHOICE: \n");

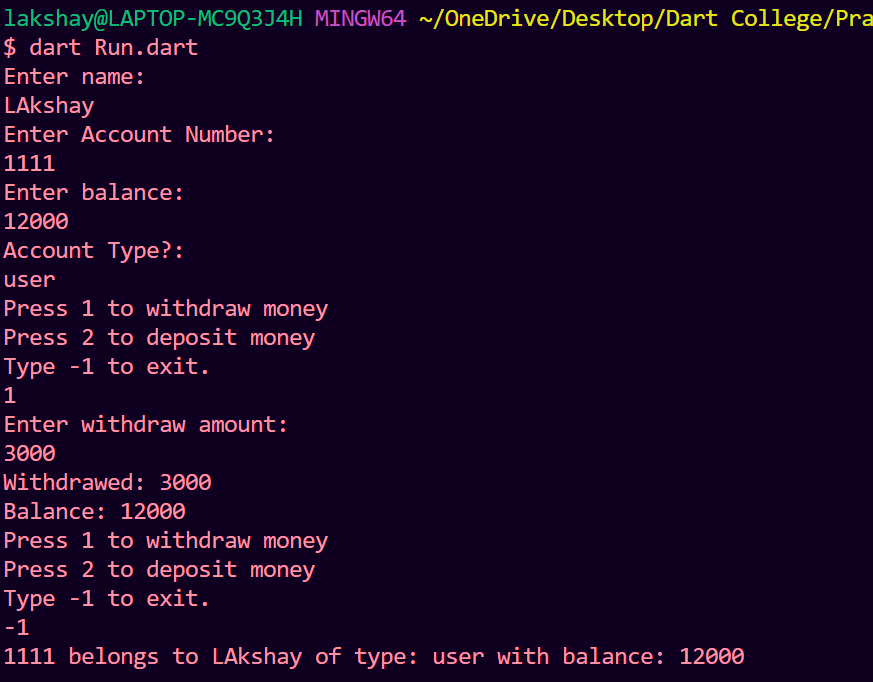
      continue;

    }

  }

  bank.display();

}

****

**Student:**

class Student {

  int? \_rollNum;

  String? \_name;

  int? \_maths;

  int? \_science;

  int? \_sst;

  int? total = 0;

  Student(int? maths, int? science, int? sst) {

*this*.\_maths = maths;

*this*.\_science = science;

*this*.\_sst = sst;

*this*.\_rollNum = 10;

*this*.\_name = "Lakshay";

  }

  int totalMarks() {

    return ((*this*.\_maths ?? 0) + (*this*.\_sst ?? 0) + (*this*.\_science ?? 0));

  }

*// rollNum getter and setter*

  set rollNum(int? rollNum) => *this*.\_rollNum = rollNum;

  int? get rollNum => \_rollNum;

*// name getter and setter*

  set name(String? name) => {*this*.\_name = name};

  String? get name => \_name;

*// maths getter and setter*

  set maths(int? maths) => *this*.\_maths = maths;

  int? get maths => \_maths;

*// science getter and setter*

  set science(int? science) => *this*.\_science = science;

  int? get science => \_science;

*// sst getter and setter*

  set sst(int? sst) => *this*.\_sst = sst;

  int? get sst => \_sst;

}

import 'dart:io';

import 'Student.dart';

void main() {

  try {

    print("Enter maths marks:");

    String? maths = stdin.readLineSync();

    int? m = int.parse("$maths");

    print("Enter science marks:");

    String? science = stdin.readLineSync();

    int? sci = int.parse("$science");

    print("Enter sst marks: ");

    String? sst = stdin.readLineSync();

    int? ss = int.parse("$sst");

    Student stu = new Student(m, sci, ss);

    int? total = stu.totalMarks();

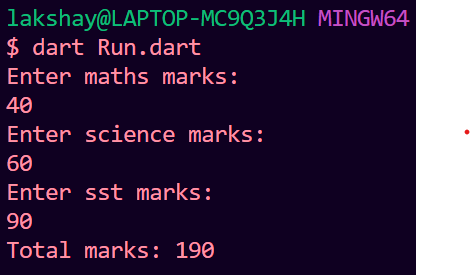
    print("Total marks: $total");

  } catch (e) {

    print("Error Occurred! \n $e");

  }

}

****

**Employee**

var allowances = {

  "teamlead": 70,

  "admin": 20,

  "technical": 60,

  "officestaff": 10

};

class Employee {

  String? \_name;

  int? \_empId;

  String? \_empDept;

  int? \_empSalary;

*// using getter and setter*

  set name(String? name) => *this*.\_name = name;

  String? get name => *this*.\_name;

  set empId(int? empId) => *this*.\_empId = empId;

  int? get empId => *this*.\_empId;

  set empDept(String? empDept) => *this*.\_empDept = empDept;

  String? get empDept => *this*.\_empDept;

  set empSalary(int? empSalary) => *this*.\_empSalary = empSalary;

  int? get empSalary => *this*.\_empSalary;

  void printDetails() {

    int? hike = allowances[empDept];

    int? total = (\_empSalary ?? 0) + ((int.parse("$hike")) \* (\_empSalary ?? 0));

    print(

        "$\_name belongs to $\_empDept with id: $\_empId Current Salary: $\_empSalary.");

    print("Hike is: $total");

  }

  Employee() {

*this*.\_empId = 0;

*this*.\_name = "sample";

*this*.\_empSalary = 50000;

*this*.\_empDept = "default";

  }

  Employee.setInputs(int id, String name, int salary, String dept) {

*this*.\_empId = id;

*this*.\_name = name;

*this*.\_empSalary = salary;

*this*.\_empDept = dept;

  }

}

import 'dart:io';

import './Employee.dart';

void main() {

  print("Enter your Department: (admin,technical,teamlead,officestaff)");

  String? dept = stdin.readLineSync();

  if (dept != null) {

    dept = dept.toLowerCase();

    Employee e = new Employee.setInputs(10, "Lakshay", 60000, dept);

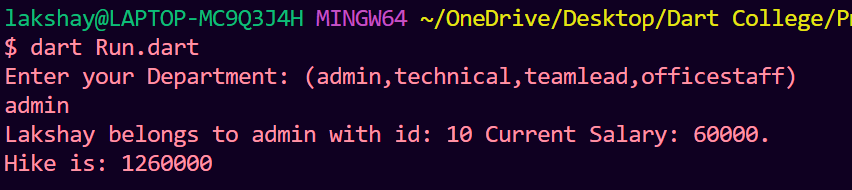
    e.printDetails();

  } else {

    print("Atleast enter one of the choices!");

  }

}

****

**Extras:**

**Target Sum**

*// Write a Dart function that takes a list of integers and a target sum as input and returns all subarrays whose sum is equal to the target sum.*

*// Input: ‘N’ = 4, ‘arr’ = [3, 1, 2, 4], 'K' = 6*

void main() {

  var input = [3, 1, 2, 4];

  int N = input.length, K = 6;

*// K is target*

  var output = [];

  int i = 0, j = 0, sum = 0;

  while (j < N) {

    sum += input[j];

    while (sum > K && i < j) {

      sum -= input[i];

      i++;

    }

    if (sum == K) {

      var o = [];

      for (int x = i; x <= j; x++) {

        o.add(input[x]);

      }

      output.add(o);

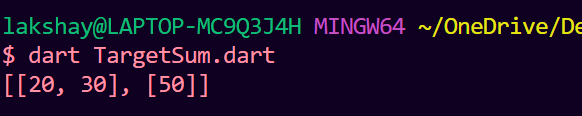
    }

    if (j < N) j++;

  }

  print(output);

}

****

**Longest Subsequence:**

*// Write a Dart function that finds the length of the longest increasing subsequence in a given list of integers.*

*// A longest increasing subsequence is a subsequence of a given sequence in which the subsequence's elements are sorted in ascending order and in which the subsequence is as long as possible. This subsequence is not necessarily contiguous or unique.*

*// Input: ‘N’ = 4, ‘arr’ = [3, 1, 2, 4]*

*// Output: 3*

void main() {

*// input*

  var input = [3, 1, 2, 4];

  int N = input.length;

  var ans = [];

  int max = 1;

  for (int i = 0; i < N - 1; i++) {

    int count = 1, last = input[i];

    var arr = [];

    arr.add(last);

    for (int j = i + 1; j < N; j++) {

      if (last < input[j]) {

        count++;

        last = input[j];

        arr.add(input[j]);

      }

    }

    if (max < count) {

      max = count;

      ans = arr;

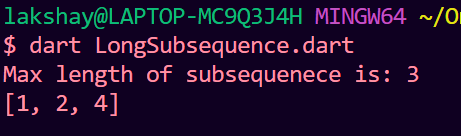
    }

  }

  print("Max length of subsequenece is: $max");

  print(ans);

}

****