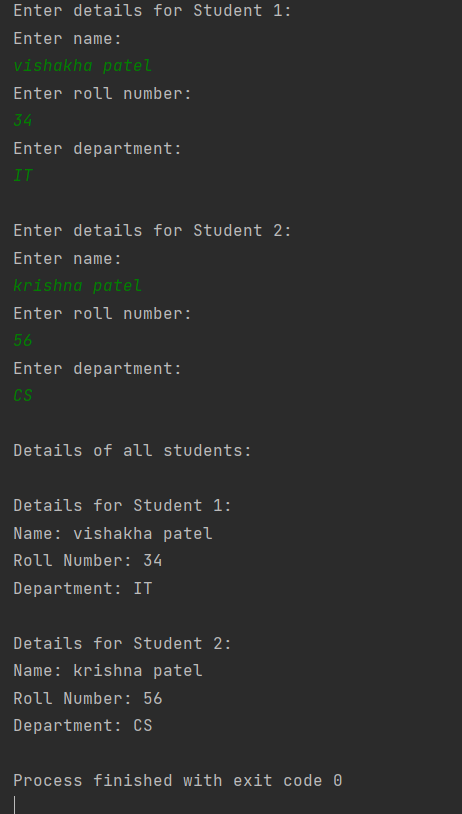
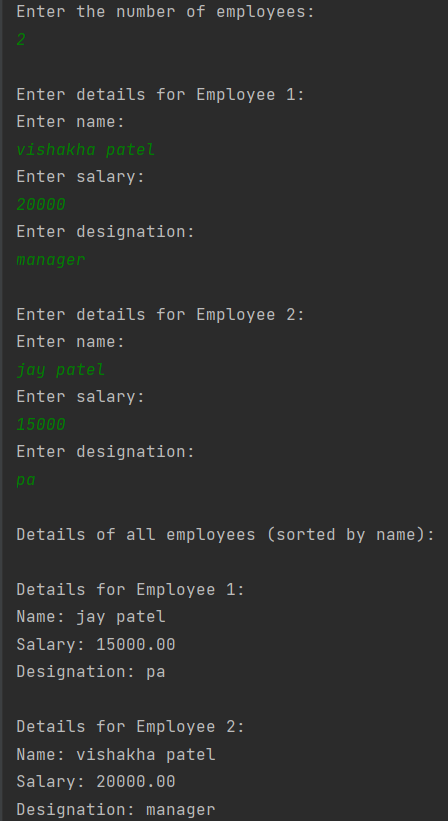
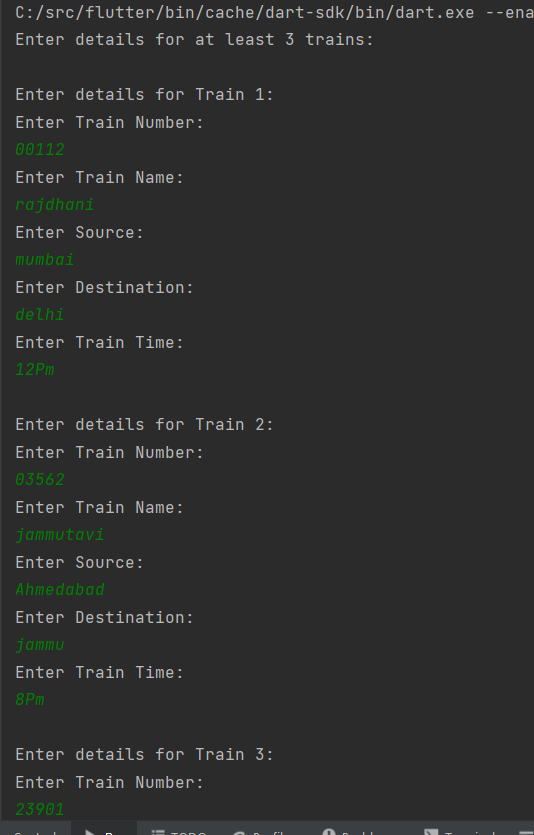
// ignore\_for\_file: avoid\_print  
  
import 'dart:io';  
  
class Student {  
 String name;  
 int rollNumber;  
 String department;  
  
 Student(this.name, this.rollNumber, this.department);  
  
 void printDetails() {  
 print('Name: $name');  
 print('Roll Number: $rollNumber');  
 print('Department: $department');  
  
 }  
}  
  
void main() {  
 print('Enter the number of students:');  
 int n = int.*parse*(stdin.readLineSync()!);  
  
 List<Student> students = [];  
  
 for (int i = 0; i < n; i++) {  
 print('\nEnter details for Student ${i + 1}:');  
 print('Enter name:');  
 String name = stdin.readLineSync()!;  
 print('Enter roll number:');  
 int rollNumber = int.*parse*(stdin.readLineSync()!);  
 print('Enter department:');  
 String department = stdin.readLineSync()!;  
  
 students.add(Student(name, rollNumber, department));  
 }  
  
 print('\nDetails of all students:');  
 for (int i = 0; i < students.length; i++) {  
 print('\nDetails for Student ${i + 1}:');  
 students[i].printDetails();  
 }  
}

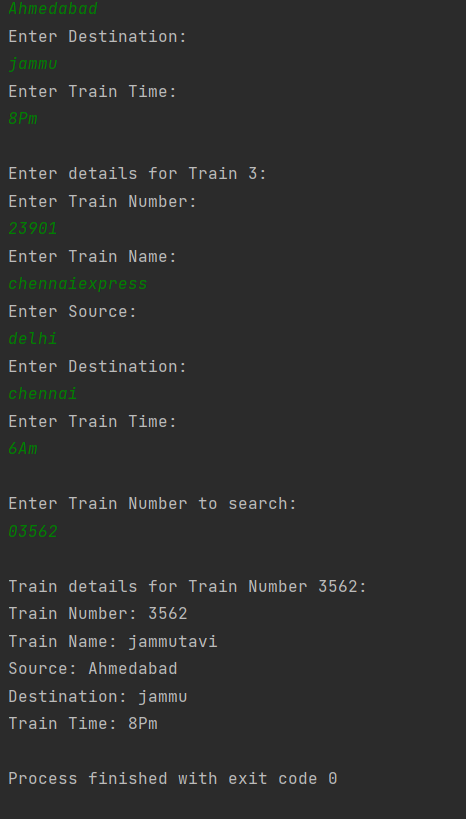


// ignore\_for\_file: avoid\_print  
  
import 'dart:io';  
  
class Employee {  
 String name;  
 double salary;  
 String designation;  
  
 Employee(this.name, this.salary, this.designation);  
  
 void printDetails() {  
 print('Name: $name');  
 print('Salary: ${salary.toStringAsFixed(2)}');  
 print('Designation: $designation');  
 }  
}  
  
void main() {  
 print('Enter the number of employees:');  
 int n = int.*parse*(stdin.readLineSync()!);  
  
 List<Employee> employees = [];  
  
 for (int i = 0; i < n; i++) {  
 print('\nEnter details for Employee ${i + 1}:');  
 print('Enter name:');  
 String name = stdin.readLineSync()!;  
 print('Enter salary:');  
 double salary = double.*parse*(stdin.readLineSync()!);  
 print('Enter designation:');  
 String designation = stdin.readLineSync()!;  
  
 employees.add(Employee(name, salary, designation));  
 }  
  
 employees.sort((a, b) => a.name.compareTo(b.name));  
  
 print('\nDetails of all employees (sorted by name):');  
 for (int i = 0; i < employees.length; i++) {  
 print('\nDetails for Employee ${i + 1}:');  
 employees[i].printDetails();  
 }  
}



// ignore\_for\_file: avoid\_print  
  
import 'dart:io';  
  
class Train {  
 int trainNumber;  
 String trainName;  
 String source;  
 String destination;  
 String trainTime;  
  
 Train(  
 {required this.trainNumber,  
 required this.trainName,  
 required this.source,  
 required this.destination,  
 required this.trainTime});  
  
 void displayDetails() {  
 print('Train Number: $trainNumber');  
 print('Train Name: $trainName');  
 print('Source: $source');  
 print('Destination: $destination');  
 print('Train Time: $trainTime');  
 }  
}  
  
void main() {  
 List<Train> trains = [];  
  
 print('Enter details for at least 3 trains:');  
 for (int i = 0; i < 3; i++) {  
 print('\nEnter details for Train ${i + 1}:');  
 print('Enter Train Number:');  
 int trainNumber = int.*parse*(stdin.readLineSync()!);  
 print('Enter Train Name:');  
 String trainName = stdin.readLineSync()!;  
 print('Enter Source:');  
 String source = stdin.readLineSync()!;  
 print('Enter Destination:');  
 String destination = stdin.readLineSync()!;  
 print('Enter Train Time:');  
 String trainTime = stdin.readLineSync()!;  
  
 trains.add(Train(  
 trainNumber: trainNumber,  
 trainName: trainName,  
 source: source,  
 destination: destination,  
 trainTime: trainTime));  
 }  
  
 print('\nEnter Train Number to search:');  
 int searchTrainNumber = int.*parse*(stdin.readLineSync()!);  
 bool found = false;  
 for (var train in trains) {  
 if (train.trainNumber == searchTrainNumber) {  
 print('\nTrain details for Train Number $searchTrainNumber:');  
 train.displayDetails();  
 found = true;  
 break;  
 }  
 }  
 if (!found) {  
 print('Train with Train Number $searchTrainNumber not found!');  
 }  
}





// ignore\_for\_file: avoid\_print  
  
import 'dart:io';  
  
class Item {  
 int itemNumber;  
 String itemName;  
 int quantity;  
 double tax;  
 double discount;  
  
 Item({  
 required this.itemNumber,  
 required this.itemName,  
 required this.quantity,  
 required this.tax,  
 required this.discount,  
 });  
  
 double calculateTotal() {  
 double total = (quantity \* (1 + tax / 100) \* (1 - discount / 100));  
 return total;  
 }  
  
 void displayDetails() {  
 print('Item Number: $itemNumber');  
 print('Item Name: $itemName');  
 print('Quantity: $quantity');  
 print('Tax: $tax%');  
 print('Discount: $discount%');  
 print('Total: ${calculateTotal().toStringAsFixed(2)}');  
 }  
}  
  
void main() {  
 String userId = 'abcd';  
 String password = '5678';  
 print('Enter User ID:');  
 String enteredUserId = stdin.readLineSync()!;  
 print('Enter Password:');  
 String enteredPassword = stdin.readLineSync()!;  
 if (enteredUserId != userId || enteredPassword != password) {  
 print('Invalid User ID or Password. Access Denied.');  
 return;  
 }  
  
 List<Item> items = [];  
  
 print('\nEnter details for items:');  
 while (true) {  
 print('\nEnter details for Item:');  
 print('Enter Item Number:');  
 int itemNumber = int.*parse*(stdin.readLineSync()!);  
 if (itemNumber == 0) {  
 break;  
 }  
 print('Enter Item Name:');  
 String itemName = stdin.readLineSync()!;  
 print('Enter Quantity:');  
 int quantity = int.*parse*(stdin.readLineSync()!);  
 print('Enter Tax:');  
 double tax = double.*parse*(stdin.readLineSync()!);  
 print('Enter Discount:');  
 double discount = double.*parse*(stdin.readLineSync()!);  
  
 items.add(Item(  
 itemNumber: itemNumber,  
 itemName: itemName,  
 quantity: quantity,  
 tax: tax,  
 discount: discount));  
 }  
  
 items.sort((a, b) => a.itemNumber.compareTo(b.itemNumber));  
  
 print('\nAll Items in Ascending Order (by Item Number):');  
 for (var item in items) {  
 item.displayDetails();  
 }  
}

