**Instructions**

* You need to solve **one programming problem** in this challenge.
* The duration of this challenge is **120 Minutes**.
* Programming questions have a **Compile and Run** option where you can run your solution against sample test cases before submitting it.
* Click **Evaluate**button only if your code compiles successfully.
* This challenge covers the following topic(s).
* Conditional constructs
* Looping
* Arrays
* Collections

**Problem Statement: Fee Reimbursement**

Due to Corona Virus situation Hudton University decides to offer **fees reimbursement scheme** to its students according to the percentage they secured in their last semester examination, as per the below criteria:

**Category A:** For 80% to 85% marks (inclusive of border values) secured in exam: Refundable amount is 40% of the fees paid by the student during the start of the academic year and a cash award.

**Category B:** For 86% to 90% marks (inclusive of border values) secured in exam: Refundable amount is 50% of the fees paid by the student during the start of the academic year and a cash award.

**Category C:** For marks above 90%: Refundable amount is 60% of the fees paid by the student during the start of the academic year and a cash award.

The University also awards the students with a **cash prize** of Rs. 3000, Rs. 5000 and Rs. 7000 for Categories A, B and C respectively.

However, for the student to be eligible for reimbursement, he **should NOT have a backlog** (arrear) in any subject.

You are tasked to create a CUI based application using C# to help the University for effectively implementing their fees reimbursement scheme.

Write a program that calculates the total amount a student receives from the University which includes refundable fee amount and cash prize, according to:

* 1. The fees he/she pays at the start of the academic year
  2. His / Her percentage of marks in the last semester exam.
  3. His/her backlog status (string: yes if there is arrear and no if there is no arrear).

**Hint:**

***Total Reimbursement = Refundable Fee Amount + Cash Prize***

**Component Specification:**

Create a class called **ReimbursementCalculator** with the below methods:

|  |  |
| --- | --- |
| **Method Name** | **Description** |
| public bool ValidateNumberOfStudents(int noOfStudents) | This method validates if the number of students is > 0 and <=10. If it returns false, print a message **“Invalid Number of Students Entered...Restart the Application”** and terminate the application. |
| public bool ValidateStudentId(int studentId) | This method validates whether the student id length is exactly 4 digits or not. If it returns false, print a message **“Student Id should be 4 Digits in Length...Restart the Application”** and terminate the application. |
| public bool ValidateFeeAmount(double feeAmount) | This method validates whether the fee is >= 25000. If it returns false, print a message **“Low Fee Amount...Restart the Application”** and terminate the application. |
| public bool ValidateMarks(int marks) | This method validates whether the marks is >=80 and <=100. If it returns false, print a message **“Less Percentage...Restart Application again”** and terminate the application. |
| public bool ContainsDuplicateId(int[] studentIds) | This method validates whether duplicate student id is present in the array. If it returns false, print a message **“Duplicate Student Ids Found...Restart the Application”** and terminate the application. |
| public Dictionary<int, double> CreateStudentRefundDetails(int noOfStudents, int[] studentIds, double[] fees, int[] marksPercentage) | This method will store student records in a Dictionary collection only **if the above 5 methods return true**. The dictionary key will be student id and value will be total refund to the student. Return the collection object to main program and display data from dictionary in the format given in sample input / output. |

Create a class called **Program** with Main method to test your application.

**Input Format:**

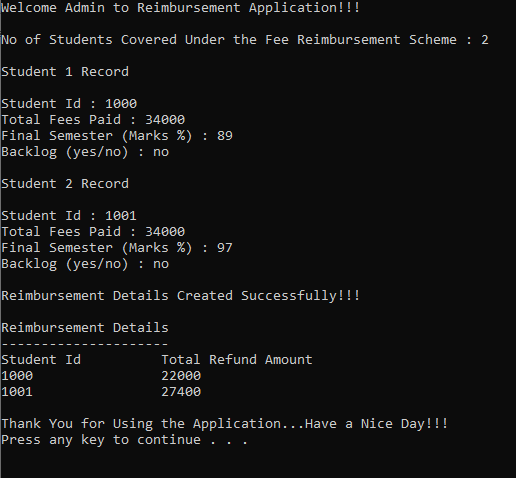
First line is an integer that denotes the Number of Students.

The next lines denote students’ ids, fees, marks and backlog information respectively.

**Output Format:**

The output will contain the key-value pair information from dictionary in which student id is the key and total refund amount is the value. While printing output there should be 20 spaces gap between student id and total refund. ***[Use the format “{0,-20}{1}”, value1,value2)]*** while printing the output to the console.

**Sample Input/Output1:**



**Sample Input/Output2:**

