# Employee Bonus

**Grade settings**: Maximum grade: 100
**Disable external file upload, paste and drop external content**: Yes
**Based on**: Employee Bonus
**Run**: Yes **Evaluate**: Yes
**Automatic grade**: Yes
A Company wants to give away bonus to its employees. You have been assigned as the programmer to automate this process. You would like to showcase your skills by creating a quick prototype. The prototype consists of the following steps:

Read Employee details from the User. The details would include id, DOB (date of birth) and salary in the given order. The datatype for id is integer, DOB is string and salary is integer.

You decide to build two hashmaps. The first hashmap contains employee id as key and DOB as value, and the second hashmap contains same employee ids as key and salary as value.

If the age of the employee in the range of 25 to 30 years (inclusive), the employee should get bonus of 20% of his salary and in the range of 31 to 60 years (inclusive) should get 30% of his salary. store the result in TreeMap in which Employee ID as key and revised salary as value. Assume the age is caculated based on the date 01-09-2014. (Typecast the bonus to integer).

**Other Rules:**

a. If Salary is less than 5000 store -100.

b. If the age is less than 25 or greater than 60 store -200.

c. a takes more priority than b i.e both if a and b are true then store -100.

You decide to write a function **calculateRevisedSalary** which takes the above hashmaps as input and returns the treemap as output. Include this function in class UserMainCode.

Create a Class Main which would be used to read employee details in step 1 and build the two hashmaps. Call the static method present in UserMainCode.

**Input and Output Format:**

Input consists of employee details. The first number indicates the size of the employees. The next three values indicate the employee id, employee DOB and employee salary. The Employee DOB format is "dd-mm-yyyy"

Output consists of a single string.

Refer sample output for formatting specifications.

**Sample Input 1:**

**2**

**1010**

**20-12-1987**

**10000**

**2020**

**01-01-1985**

**14400**

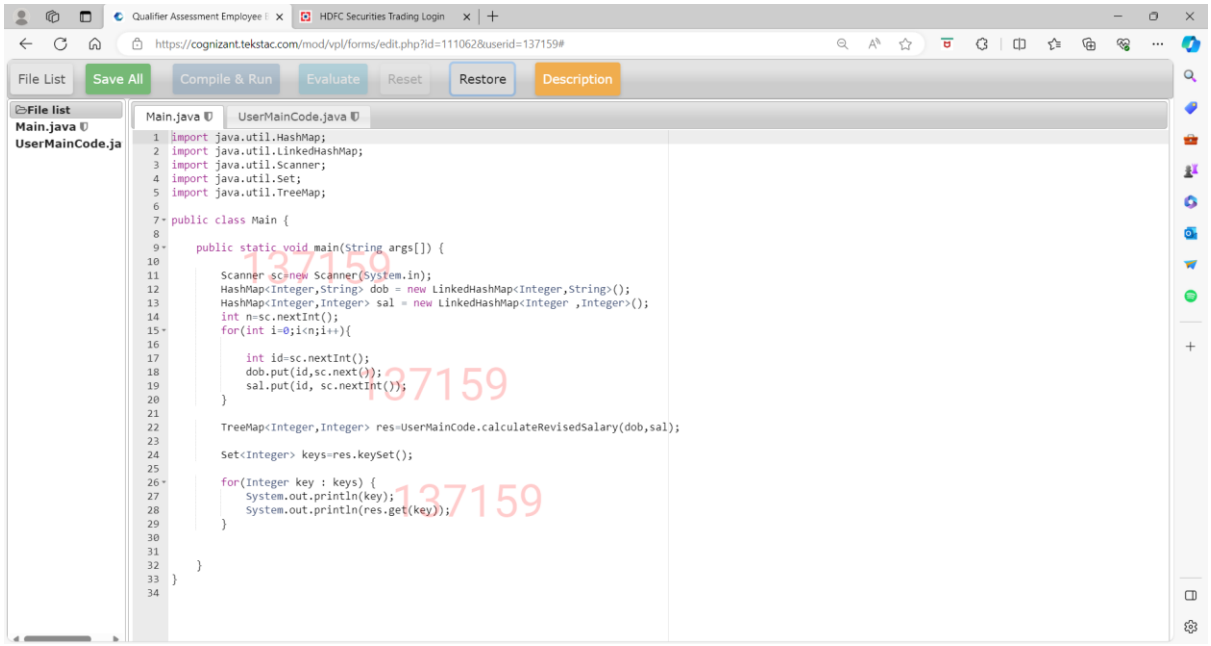
**Sample Output 1:**

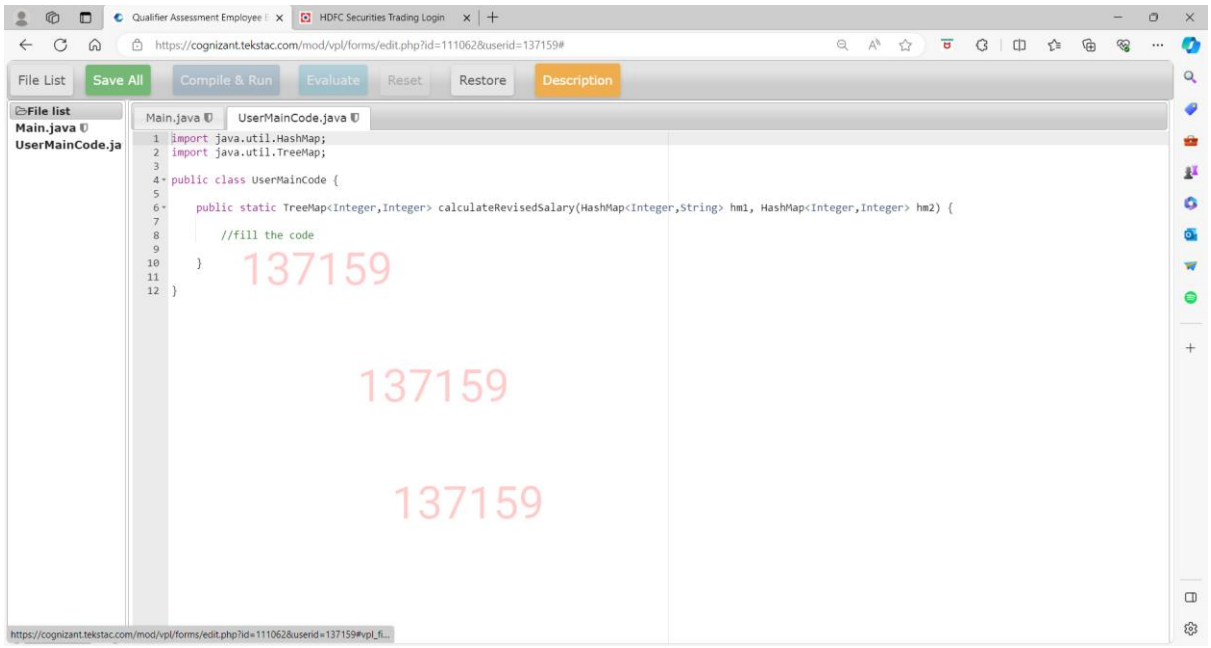
1010

12000

2020

17280


Qualifier Assessment Employee
HDFC Securities Trading Login
https://cognizant.tekstac.com/mod/vpl/forms/edit.php?id=111062&userid=137159#
File List
Save All
Compile & Run
Evaluate
Reset
Restore
Description
File list
Main.java
UserMainCode.ja
Main.java
UserMainCode.java
import java.util.HashMap;
import java.util.LinkedHashMap;
import java.util.Scanner;
import java.util.Set;
import java.util.TreeMap;
public class Main {
public static void main(String args[]) {
Scanner sc=new Scanner(System.in);
HashMap<Integer,String> dob = new LinkedHashMap<Integer,String>();
HashMap<Integer,Integer> sal = new LinkedHashMap<Integer ,Integer>();
int n=sc.nextInt();
for(int i=0;i<n;i++){
int id=sc.nextInt();
dob.put(id,sc.next());
sal.put(id, sc.nextInt());
}
TreeMap<Integer,Integer> res=UserMainCode.calculateRevisedSalary(dob,sal);
Set<Integer> keys=res.keySet();
for(Integer key : keys) {
System.out.println(key);
System.out.println(res.get(key));
}
}
}
137159
137159
137159


Qualifier Assessment Employee
HDFC Securities Trading Login
https://cognizant.tekstac.com/mod/vpl/forms/edit.php?id=111062&userid=137159#
File List
Save All
Compile & Run
Evaluate
Reset
Restore
Description
File list
Main.java
UserMainCode.ja
Main.java
UserMainCode.java
import java.util.HashMap;
import java.util.TreeMap;
public class UserMainCode {
public static TreeMap<Integer,Integer> calculateRevisedSalary(HashMap<Integer,String> hm1, HashMap<Integer,Integer> hm2) {
//fill the code
}
}
137159
137159
137159
https://cognizant.tekstac.com/mod/vpl/forms/edit.php?id=111062&userid=137159#vpl_fi...