**Do Select Coding Questions**

**Q11 - Answer**

package doselectquestions;

class InvalidDaysException extends Exception

{

public InvalidDaysException(String str)

{

super(str);

}

}

class InvalidSalaryException extends Exception

{

public InvalidSalaryException(String str)

{

super(str);

}

}

class SalaryData

{

public String name;

public int daysInMonth;

public double salary;

public SalaryData(String name,

int daysInMonth,double salary)

{

this.name = name;

this.daysInMonth = daysInMonth;

this.salary = salary;

}

}

class Validator

{

public String validSalaryData(SalaryData s) throws

InvalidDaysException, InvalidSalaryException

{

if((s.daysInMonth>0 && s.daysInMonth<28 )|| s.daysInMonth==29)

{

throw new InvalidDaysException("Invalid Days");

}

else if (s.salary <0 || s.salary > 1000000)

{

throw new InvalidSalaryException("Invalid Salary");

}

else

{

return "Valid Data";

}

}

public double totalSalary(SalaryData s)

{

return s.daysInMonth\*s.salary;

}

}

public class salaryvalidatorproblemeleven {

public static void main(String[] args) throws

InvalidDaysException, InvalidSalaryException

{

try

{

SalaryData s = new SalaryData("Steve",28,10000);

Validator val = new Validator();

String ans = val.validSalaryData(s);

double sal =0.0;

if(ans == "Valid Data")

{

sal += val.totalSalary(s);

}

System.*out*.println("ans = "+ans);

System.*out*.println("sal = "+sal);

}

catch (InvalidDaysException e)

{

System.*out*.println(e.getMessage());

}

catch (InvalidSalaryException e)

{

System.*out*.println(e.getMessage());

}

}}