OOP Lab7

1- Define Class **Duration**

To include Three Attributes (**Hours**, **Minutes**, **Seconds**)

Output from **Print** Should follow this pattern

Hours: 1, Minutes :30, Seconds :20

Support All Required Constructors to Produce this output

Duration D1 =new Duration (1,10,15);

D1.Print();

Output: Hours: 1, Minutes :10 , Seconds :15

Duration D2 =new Duration (3600);

D2.Print();

Output: Hours: 1, Minutes :0 , Seconds :0

Duration D3 =new Duration (7800);

D3.Print();

Output: Hours: 2, Minutes :10 , Seconds :0

Duration D4 =new Duration (666);

D.Print();

Output: Minutes :11 , Seconds :6

Implement All required Operators overloading's to enable this Code

D3=D1+D2

D3=D1 + 7800

If (D1>D2);

If (D1<=D2);

2- Repeat task 2 from the previous lab (OOP Lab6) with abstract shape class and abstract CalcArea method in shape class and override it in all derived class then update calcallareas method in class program

3- Create abstract class Employee which contains ID , Name , Salary , array of Client (create Client class which contains ID , Name , City and ShowClientDetails Method) ,abstract ShowEmployeeDetails method which will print all data about caller Employee object
   - Create HREmployee class which will inherit from Employee
   - Create PREmployee class which will inherit from Employee
   - In main create an array of Employees which will contains HREmployees and PREmployees objects , then pass this array to static method GetEmployeeWithLongestClientsArray which will return an Employee Object
   - Try to run the following statements
     HREmployee hrone = new HREmployee();// complete the object creation
     HREmployee hrtwo = new HREmployee();// complete the object creation
     HREmployee hrthree=hrone + hrtwo;
     If(hrone>hrtwo)
     If(hrone>=hrtwo)

4- Create your Authenticate Class which will contains loginMethod , forgetPassword , ResetPassword methods , property from type student , student class will contains ID , Name , username , userpass , email , print method
   - Develop loginMethod to accept (username , userpass,useremail) , (useremail,userpass) , (id,userpass) , (id,email) in all overloads will return bool value if passed parameters matched with student object data otherwise will return false;
   - Develop forgetPassword to accept (username) , (useremail,id) , (userid) Which will return userpass if passed values matched with student object otherwise return “user not found !”
   - Develop ResetPassWord to accept
     (id,oldpassword,newpassword) ,
     (useremail,oldpassword,newpassword),

(userid,username,oldpassword,newpassword)
Which will change current student password with new one if passed data matched and return true if it succeeded otherwise return false;