



```
System.out.println("enter employee id");
emp_id=sc.nextInt();
//System.out.println(+e_id);
```

```
ResultSet rs = stmt.executeQuery("select esalary from employeee where emp_id="+emp_id);
```

```
while(rs.next()) {
    emp_sal=rs.getInt(1);
    System.out.println(emp_sal);
    //conditions
    if(emp_sal>=100.00 && emp_sal<1500) {
        float pf =(emp_sal*5)/100;
        System.out.println("your pf is :" +pf);
    }
    if(emp_sal>=1500.00 && emp_sal<2000) {
        float pf =(emp_sal*6)/100;
        System.out.println("your pf is :" +pf);
    }
    if(emp_sal>=2000.00) {
        float pf =(emp_sal*7)/100;
        System.out.println("your pf is :" +pf);
    }
}
```

```
}
```

Location	Resource	Path	Location	Type
Writable			Smart Insert	149 : 1 : 4765



```
package lab_1st_dec;

import java.sql.Connection;

public class Helper {
    static {
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");

        } catch (Exception e) {
            System.out.println(e);
        }
    }
    public static Connection con() throws SQLException{
        return DriverManager.getConnection("jdbc:mysql://localhost:3306/c2268","root","root");
    }
    public static void main(String[] args) throws SQLException { //main method
        DuplicateDataException eu= new DuplicateDataException();//creating object of main class
        Scanner scan=new Scanner(System.in); //scanner class
        System.out.println("""
        Press 1 for Insert Department Records :
        Press 2 for Insert Employee Records :
        Press 3 for Display Records :
        Press 4 for checking PF :
        Press 5 for Exit :
        """);
    }
}
```





EmployeeUploader.java Helper.java X DuplicateDataException.j... Main_employeeUploader.j... Id_Not_foundException.j... EmployeeUploader.java

Press 4 for checking PF :

Press 5 for Exit :

""");

`int variable = scan.nextInt();``System.out.println("-----");`

//cases

`switch(variable) {``case 1 :``eu.storeDepartmentDetails();``break;``case 2 :``try {``eu.storeEmployeeDetails();``} catch (SQLException | DuplicateDataException |``LessSalaryAmountException``| DepartmentNotFoundException e) {``System.out.println(e);;``}``break;``case 3 :``eu.retrieveDeptdetails();``break;``case 4 :``//System.out.println("Enter employee id");``//int id= scan.nextInt();``eu.calculatePF();``break;`

blems X @ Javadoc Declaration

, 154 warnings, 0 others (Filter matched 162 of 216 items)

tion	Resource	Path	Location	Type	Smart Insert	15:35:289
			Writable			





EmployeeUploader.java Helper.java X DuplicateDataException.j... Main_employeeUploader.j... Id_Not_foundException.j... EmployeeUploader.java

```
switch(variable) {  
case 1 :  
    eu.storeDepartmentDetails();  
    break;  
case 2 :  
    try {  
        eu.storeEmployeeDetails();  
    } catch (SQLException | DuplicateDataException |  
        LessSalaryAmountException  
        | DepartmentNotFoundException e) {  
        System.out.println(e);  
    }  
    break;  
case 3 :  
    eu.retrieveDeptdetails();  
    break;  
case 4 :  
    //System.out.println("Enter employee id");  
    //int id= scan.nextInt();  
    eu.calculatePF();  
    break;  
case 5 :  
    System.out.println("thank you");  
    System.exit(0);  
    break;  
}  
}}
```

blems X @ Javadoc Declaration

, 154 warnings, 0 others (Filter matched 162 of 216 items)

Location	Resource	Path	Location	Type
Writable			Smart Insert	15 : 35 : 289





```
package lab_1st_dec;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.DuplicateFormatFlagsException;
import java.util.Scanner;

import com.sms.Helper;
//validation
class DuplicateDataException extends Exception{
    public DuplicateDataException(String str) {
        super(str);
    }
}
class LessSalaryAmountException extends Exception{
    public LessSalaryAmountException(String str) {
        super(str);
    }
}
class DepartmentNotFoundException extends Exception{
    public DepartmentNotFoundException(String str) {
        super(str);
    }
}
```

Location	Resource	Path	Location	Type
Writable	Smart Insert	116 : 9 : 3935		



```
public class EmployeeUploader {
    Scanner sc=new Scanner(System.in); //taking scanner class
    int dep_id;
    String dep_name, dep_head, dep_des;
    //store
    public void storeDepartmentDetails() throws SQLException{ //1st method
        //taking user input to store data
        System.out.println("enter department id");
        dep_id=sc.nextInt();
        System.out.println("enter department name ");
        dep_name=sc.next();
        System.out.println("enter department head ");
        dep_head=sc.next();
        System.out.println("enter department description ");
        dep_des=sc.next();
        Connection conn=Helper.con(); //connection with helper class
        PreparedStatement stmt=conn.prepareStatement("insert into Department values(?,?,?,?)");
        stmt.setInt(1, dep_id);
        stmt.setString(2, dep_name);
        stmt.setString(3, dep_head);
        stmt.setString(4, dep_des);
        stmt.executeUpdate();
        ResultSet rs=stmt.executeQuery("select * from Department");
        //iteration
        while(rs.next()) {
            System.out.println(rs.getInt(1)+" "+rs.getString(2)+" "+rs.getString(3)+" "+
```



```
rs.getString(4));
}
}
int emp_id,emp_sal,emp_con_no;
String emp_name,emp_add;
//2nd method
public void storeEmployeeDetails() throws SQLException,DuplicateDataException,
LessSalaryAmountException, DepartmentNotFoundException{
    //storing data from user
    System.out.println("enter Employee id: ");
    emp_id=sc.nextInt();
    System.out.println("enter Employee name: ");
    emp_name=sc.next();
    System.out.println("enter Employee address: ");
    emp_add=sc.next();
    System.out.println("enter Employee salary: ");
    emp_sal=sc.nextInt();
    System.out.println("enter Employee contact no: ");
    emp_con_no=sc.nextInt();
    System.out.println("enter Employee d_id");
    dep_id=sc.nextInt();
    Connection conn=Helper.con();//connection
    PreparedStatement stmt=conn.prepareStatement("insert into Employee values(?,?,?,?,?,?)");
    stmt.setInt(1, emp_id);
    stmt.setString(2, emp_name);
    stmt.setString(3, emp_add);
    stmt.setInt(4, emp_sal);
```




```
stmt.setInt(5, emp_con_no);
stmt.setInt(6, dep_id);
stmt.executeUpdate();
//conditions
if((dep_id==emp_id) || (dep_name==emp_name)){
    throw new DuplicateFormatFlagsException("employee already exists");
}
if(emp_sal<1000) {
    throw new LessSalaryAmountException("salary cannot be less than 1000");
}
if(dep_id==0) {
    throw new DepartmentNotFoundException("department does not exists");
}

System.out.println("successfully inserted");

}

public void retrieveDeptdetails() throws SQLException{//3rd method
    System.out.println("Enter the dept id you want to retrieve details "
        + "from employee and department");
    int dep_id=sc.nextInt();
    Connection conn=Helper.con();//connection
    Statement stmt=conn.createStatement();
    ResultSet rs=stmt.executeQuery("select * from Employee");
    //iteration for employee details
    while(rs.next()) {
```

Location	Resource	Path	Location	Type
Writable	Smart Insert	116 : 9 : 3935		





```
        System.out.println(rs.getInt(1)+" "+rs.getString(2)+" "+rs.getString(3)+" "+
rs.getInt(4)+" "+rs.getInt(5)+" "+rs.getInt(6));
    }
    ResultSet rs1=stmt.executeQuery("select * from Department");
    //iteration for department details
    while(rs1.next()) {
        System.out.println(rs1.getInt(1)+" "+rs1.getString(2)+" "+rs1.getString(3)+" "
+rs1.getString(4));
    }
}

public void calculatePF() throws SQLException{
    Connection conn =Helper.con();
    Statement stmt =conn.createStatement();
    System.out.println("enter employee id");
    emp_id=sc.nextInt();
    //System.out.println(+e_id);

    ResultSet rs = stmt.executeQuery("select esalary from employeee where emp_id="+emp_id);
    while(rs.next()) {
        emp_sal=rs.getInt(1);
        System.out.println(emp_sal);
        //conditions
        if(emp_sal>=100.00 && emp_sal<1500) {
            float pf =(emp_sal*5)/100;
            System.out.println("your pf is :" +pf);
        }
    }
}
```

tion	Resource	Path	Location	Type	
			Writable	Smart Insert	116 : 9 : 3935