

Nama : Faiqah Farras F.

Kelas : 4IA12

NPM : 50420440

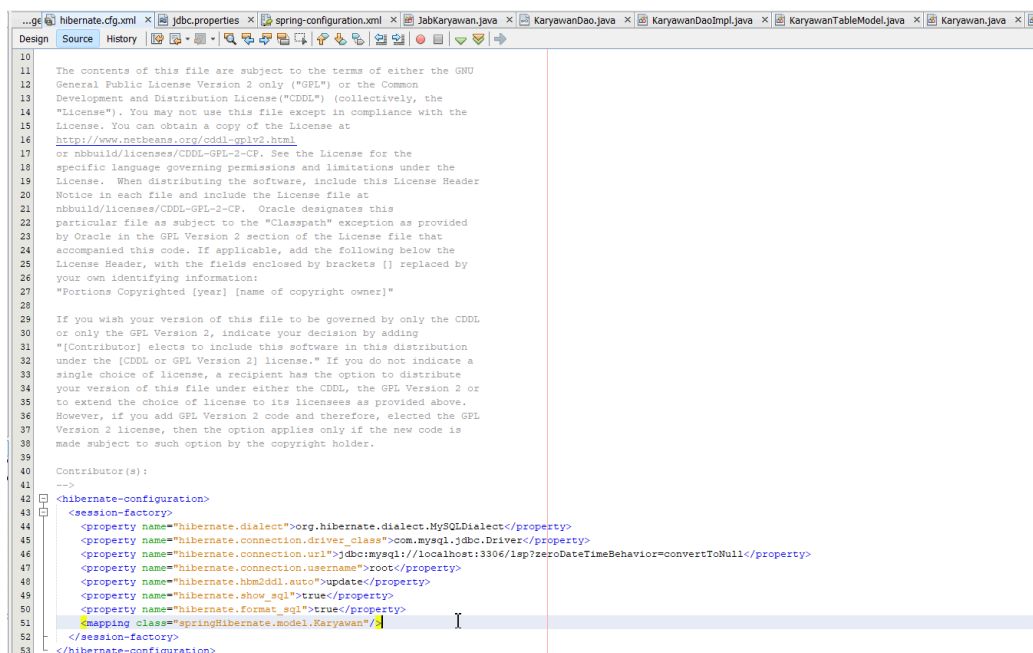
Tugas 3.

Buatlah program pengolahan transaksi yang menampilkan perhitungan total gaji yang diterima karyawan dalam bentuk slip gaji berdasarkan data yang menjadi input. (Gunakan 5 Data, 1 Manager, 1 Supervisor, 3 Staff). Buatlah program yang digunakan untuk pengolahan transaksi seperti pencetakan laporan gaji karyawan per bulan. (dalam bentuk file atau dokumen tercetak). Dalam proses menjalankan program dapat diperlihatkan error dari aplikasi yang anda buat.

Buatlah program yang untuk menambah data karyawan. Tambahkan 3 data karyawan dengan posisi 2 staff 1 supervisor. Buat program untuk menampilkan laporan gaji karyawan untuk periode 2 bulan sebelumnya. Maka anda wajib membuat data untuk proses bulan Maret, April, Mei).

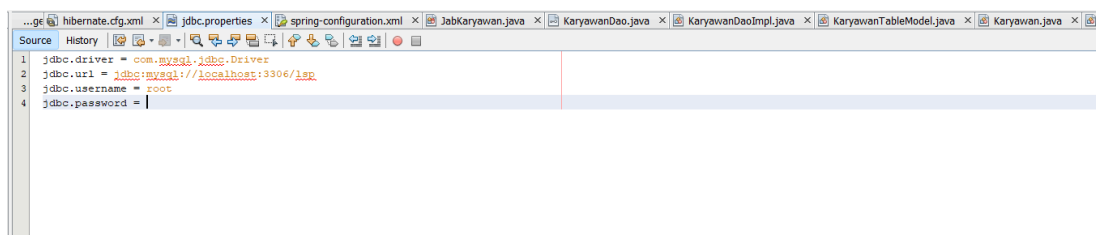
Source Code.

- hibernate.cfg



```
10
11 The contents of this file are subject to the terms of either the GNU
12 General Public License Version 2 only ("GPL") or the Common
13 Development and Distribution License ("CDDL") (collectively, the
14 "License"). You may not use this file except in compliance with the
15 License. You can obtain a copy of the License at
16 http://www.netbeans.org/cddl-gplv2.html
17 or nbbuild/licenses/CDDL-GPL-2-CP. See the License for the
18 specific language governing permissions and limitations under the
19 License. When distributing the software, include this License Header
20 Notice in each file and include the License file at
21 nbbuild/licenses/CDDL-GPL-2-CP. Oracle designates this
22 particular file as subject to the "Classpath" exception as provided
23 by Oracle in the GPL Version 2 section of the License file that
24 accompanied this code. If applicable, add the following below the
25 License Header, with the fields enclosed by brackets [] replaced by
26 your own identifying information:
27 "Portions Copyrighted [year] [name of copyright owner]"
28
29 If you wish your version of this file to be governed by only the CDDL
30 or only the GPL Version 2, indicate your decision by adding
31 "[Contributor] elects to include this software in this distribution
32 under the [CDDL or GPL Version 2] license." If you do not indicate a
33 single choice of license, a recipient has the option to distribute
34 your version of this file under either the CDDL, the GPL Version 2 or
35 to extend the choice of license to its licensees as provided above.
36 However, if you add GPL Version 2 code and therefore, elected the GPL
37 Version 2 license, then the option applies only if the new code is
38 made subject to such option by the copyright holder.
39
40 Contributor(s):
41 -->
42
43 <hibernate-configuration>
44 <!--
45 <property name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>
46 <property name="hibernate.connection.driver_class">com.mysql.jdbc.Driver</property>
47 <property name="hibernate.connection.url">jdbc:mysql://localhost:3306/lsp?zeroDateTimeBehavior=convertToNull</property>
48 <property name="hibernate.connection.username">root</property>
49 <property name="hibernate.hbm2ddl.auto">update</property>
50 <property name="hibernate.show_sql">true</property>
51 <property name="hibernate.format_sql">true</property>
52 <mapping class="springhibernate.model.karyawan"/>
53 </hibernate-configuration>
```

- jdbc.properties



```
1 jdbc.driver = com.mysql.jdbc.Driver
2 jdbc.url = jdbc:mysql://localhost:3306/lsp
3 jdbc.username = root
4 jdbc.password =
```

- spring-configuration.xml

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <beans xmlns="http://www.springframework.org/schema/beans"
3       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4       xmlns:context="http://www.springframework.org/schema/context"
5       xmlns:p="http://www.springframework.org/schema/p"
6       xmlns:tx="http://www.springframework.org/schema/tx"
7       xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans-4.0.xsd
8                           http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context-4.0.xsd
9                           http://www.springframework.org/schema/tx http://www.springframework.org/schema/tx/spring-tx-4.0.xsd"
10  >
11
12
13
14  <context:annotation-config/>
15  <tx:annotation-driven transaction-manager="transactionManager"/>
16
17  <context:component-scan base-package="springHibernate"/>
18  <context:property-placeholder location="classpath:jdbc.properties"/>
19
20  <bean id="dataSource" class="org.springframework.jdbc.datasource.DriverManagerDataSource"
21        p:driverClassName="${jdbc.driver}"
22        p:url="${jdbc.url}"
23        p:username="${jdbc.username}"
24        p:password="${jdbc.password}"
25  >
26
27
28  <bean id="sessionFactory" class="org.springframework.orm.hibernate4.LocalSessionFactoryBean"
29        p:dataSource-ref="dataSource"
30        p:configLocation="classpath:hibernate.cfg.xml"
31  >
32
33  <bean id="transactionManager" class="org.springframework.orm.hibernate4.HibernateTransactionManager"
34        p:sessionFactory-ref="sessionFactory"
35  >
36
37  </beans>

```

- App.java

```

1 /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6 package springHibernate;
7
8 import org.springframework.context.ApplicationContext;
9 import org.springframework.context.support.ClassPathXmlApplicationContext;
10 import springHibernate.view.KaryawanView;
11 import springHibernate.service.KaryawanService;
12
13 /**
14  *
15  * @author hp
16  */
17 public class App {
18     private static ApplicationContext applicationContext;
19     /**
20      * @param args the command line arguments
21      */
22     public static void main(String[] args) {
23         applicationContext = new ClassPathXmlApplicationContext("classpath:spring-configuration.xml");
24         new KaryawanView().setVisible(true);
25     }
26
27     public static KaryawanService getKaryawanService() {
28         return (KaryawanService) applicationContext.getBean("KaryawanService");
29     }
30
31 }

```

- Koneksi.java

```

1 /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6 package springHibernate;
7
8 import com.mysql.jdbc.Connection;
9 import com.mysql.jdbc.jdbc2.optional.MysqlDataSource;
10 import java.sql.SQLException;
11
12 /**
13  *
14  * @author faisqah
15  */
16 public class Koneksi {
17     static Connection c;
18
19     public static Connection connection() {
20         if (c == null) {
21             MysqlDataSource data = new MysqlDataSource();
22             data.setDatabaseName("lap");
23             data.setUser("root");
24             data.setPassword("");
25             try {
26                 c = (Connection) data.getConnection();
27             } catch (SQLException ex) {
28                 ex.printStackTrace();
29             }
30             return c;
31         }
32     }
33 }

```

- KaryawanTableModel.java

```

1  /*
2   * To change this license header, choose License Headers in Project Properties.
3   * To change this template file, choose Tools | Templates
4   * and open the template in the editor.
5   */
6   package springhibernate.configuration;
7
8   import java.util.ArrayList;
9   import java.util.List;
10  import javax.swing.table.AbstractTableModel;
11  import springhibernate.model.Karyawan;
12  import springhibernate.model.JabKaryawan;
13
14  /**
15   *
16   * @author faiqah
17   */
18  public class KaryawanTableModel extends AbstractTableModel {
19
20      private List<Karyawan> karyawans = new ArrayList<>();
21      private final String HEADER[] = {"NIP", "Nama Karyawan",
22          "Alamat", "Jabatan", "Tanggal Lahir", "Tanggal Mulai Kerja", "Gaji Pokok", "Bonus", "PPH", "Total Gaji"};
23
24      public KaryawanTableModel(List<Karyawan> karyawans) {
25          this.karyawans = karyawans;
26          hitungGaji();
27      }
28
29      @Override
30      public int getRowCount() {
31          return karyawans.size();
32      }
33
34      @Override
35      public int getColumnCount() {
36          return HEADER.length;
37      }
38
39      @Override
40      public String getColumnName(int columnIndex) {
41          return HEADER[columnIndex];
42      }

```

```

43
44      @Override
45      public Object getValueAt(int rowIndex, int columnIndex) {
46          Karyawan karyawan = karyawans.get(rowIndex);
47
48          switch (columnIndex) {
49              case 0:
50                  return karyawan.getNipKaryawan();
51              case 1:
52                  return karyawan.getNamaKaryawan();
53              case 2:
54                  return karyawan.getAlmKaryawan();
55              case 3:
56                  return karyawan.getJabKaryawan();
57              case 4:
58                  return karyawan.getTglLahir();
59              case 5:
60                  return karyawan.getTglKerja();
61              case 6:
62                  return karyawan.getGajiKaryawan();
63              case 7:
64                  return karyawan.getBonus();
65              case 8:
66                  return karyawan.getPph();
67              case 9:
68                  return karyawan.getTotalGaji();
69              default:
70                  return null;
71          }
72      }
73
74      // Method untuk menghitung gaji berdasarkan jabatan
75      private void hitungGaji() {
76          for (Karyawan karyawan : karyawans) {
77              karyawan.hitungGaji();
78          }
79      }
80
81  }

```

- Karyawan Controller.java

```

1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6  package springhibernate.controller;
7
8  import java.text.ParseException;
9  import java.text.SimpleDateFormat;
10 import java.util.List;
11 import java.util.logging.Level;
12 import java.util.logging.Logger;
13 import javax.swing.JOptionPane;
14 import springhibernate.App;
15 import springhibernate.configuration.KaryawanTableModel;
16 import springhibernate.model.Karyawan;
17 import springhibernate.model.JabKaryawan;
18 import springhibernate.view.KaryawanView;
19
20 /**
21 *
22 * @author hp
23 */
24 public class KaryawanController {
25     private final KaryawanView karyawanView;
26     private KaryawanTableModel karyawanTableModel;
27     private List<Karyawan> karyawans;
28
29     public KaryawanController(KaryawanView karyawanView) {
30         this.karyawanView = karyawanView;
31     }
32
33     public void tampilData() {
34         karyawans = App.getKaryawanService().getKaryawans();
35         karyawanTableModel = new KaryawanTableModel(karyawans);
36         this.karyawanView.getTabel().setModel(karyawanTableModel);
37     }
38
39     public void show() {
40         int index = this.karyawanView.getTabel().getSelectedRow();
41
42         this.karyawanView.getNipKaryawan().setText(String.valueOf(this.karyawanView.getTabel().getValueAt(index, 0)));
43         this.karyawanView.getNamaKaryawan().setText(String.valueOf(this.karyawanView.getTabel().getValueAt(index, 1)));
44         this.karyawanView.getAlmKaryawan().setText(String.valueOf(this.karyawanView.getTabel().getValueAt(index, 2)));
45         this.karyawanView.getJabKaryawan().setText(String.valueOf(this.karyawanView.getTabel().getValueAt(index, 3)));

```

```

46         this.karyawanView.getTglLahir().setText(String.valueOf(this.karyawanView.getTabel().getValueAt(index, 4)));
47         this.karyawanView.getTglKerja().setText(String.valueOf(this.karyawanView.getTabel().getValueAt(index, 5)));
48     }
49
50     public void clear() {
51         this.karyawanView.getNipKaryawan().setText("");
52         this.karyawanView.getNamaKaryawan().setText("");
53         this.karyawanView.getAlmKaryawan().setText("");
54         this.karyawanView.getJabKaryawan().setText("");
55         this.karyawanView.getTglLahir().setText("");
56         this.karyawanView.getTglKerja().setText("");
57     }
58
59     public void saveKaryawan() {
60         Karyawan karyawan = new Karyawan();
61         karyawan.setNipKaryawan(Integer.parseInt(this.karyawanView.getNipKaryawan().getText()));
62         karyawan.setNamaKaryawan(this.karyawanView.getNamaKaryawan().getText());
63         karyawan.setAlmKaryawan(this.karyawanView.getAlmKaryawan().getText());
64         karyawan.setJabKaryawan(JabKaryawan.valueOf(this.karyawanView.getJabKaryawan().getText().toUpperCase()));
65         SimpleDateFormat dateFormat = new SimpleDateFormat("yyyy-MM-dd");
66
67         try {
68             karyawan.setTglLahir(dateFormat.parse(this.karyawanView.getTglLahir().getText()));
69             karyawan.setTglKerja(dateFormat.parse(this.karyawanView.getTglKerja().getText()));
70         } catch (ParseException ex) {
71             Logger.getLogger(KaryawanController.class.getName()).log(Level.SEVERE, null, ex);
72         }
73
74         App.getKaryawanService().save(karyawan);
75         karyawan.hitungGaji();
76         JOptionPane.showMessageDialog(null, "Data Berhasil Disimpan", "Info", JOptionPane.INFORMATION_MESSAGE);
77         clear();
78         tampilData();
79     }
80
81     public void updateKaryawan() {
82         Karyawan karyawan = new Karyawan();
83         karyawan.setNipKaryawan(Integer.parseInt(this.karyawanView.getNipKaryawan().getText()));
84         karyawan.setNamaKaryawan(this.karyawanView.getNamaKaryawan().getText());
85         karyawan.setAlmKaryawan(this.karyawanView.getAlmKaryawan().getText());
86         karyawan.setJabKaryawan(JabKaryawan.valueOf(this.karyawanView.getJabKaryawan().getText().toUpperCase()));
87         SimpleDateFormat dateFormat = new SimpleDateFormat("yyyy-MM-dd");
88

```

```

Source History
89         try {
90             karyawan.setTglLahir(dateFormat.parse(this.karyawanView.getTglLahir().getText()));
91             karyawan.setTglKerja(dateFormat.parse(this.karyawanView.getTglKerja().getText()));
92         } catch (ParseException ex) {
93             Logger.getLogger(KaryawanController.class.getName()).log(Level.SEVERE, null, ex);
94         }
95
96         App.getKaryawanService().update(karyawan);
97         karyawan.hitungGaji();
98         JOptionPane.showMessageDialog(null, "Data Berhasil Diedit", "Info", JOptionPane.INFORMATION_MESSAGE);
99         clear();
100        tampilData();
101    }
102
103    public void deleteKaryawan() {
104        if (this.karyawanView.getNipKaryawan().getText().isEmpty()) {
105            JOptionPane.showMessageDialog(null, "Karyawan belum dipilih", "Error", JOptionPane.ERROR_MESSAGE);
106        } else {
107            int pilih = JOptionPane.showConfirmDialog(null, "Apakah data ingin dihapus?", "Warning", JOptionPane.YES_NO_OPTION, JOptionPane.WARNING_MESSAGE);
108            if (pilih == JOptionPane.YES_OPTION) {
109                int nipKaryawan = Integer.parseInt(this.karyawanView.getNipKaryawan().getText());
110                App.getKaryawanService().delete(nipKaryawan);
111                JOptionPane.showMessageDialog(null, "Data Berhasil Dihapus", "Info", JOptionPane.INFORMATION_MESSAGE);
112                clear();
113                tampilData();
114            }
115        }
116    }
117
118    }

```

I

• KaryawanDao.java

```

Source History
1  /*
2   * To change this license header, choose License Headers in Project Properties.
3   * To change this template file, choose Tools | Templates
4   * and open the template in the editor.
5   */
6  package springhibernate.dao;
7  import java.util.List;
8  import springhibernate.model.Karyawan;
9
10     /**
11     *
12     * @author hp
13     */
14     public interface KaryawanDao {
15         public void save(Karyawan karyawan);
16         public void update(Karyawan karyawan);
17         public void delete(Karyawan karyawan);
18         public Karyawan getKaryawan(String nipKaryawan);
19         public List<Karyawan> getKaryawans();
20     }

```

• KaryawanDaoImpl.java

```

Source History
1  /*
2   * To change this license header, choose License Headers in Project Properties.
3   * To change this template file, choose Tools | Templates
4   * and open the template in the editor.
5   */
6  package springhibernate.dao;
7  import java.util.List;
8  import springhibernate.model.Karyawan;
9  import org.hibernate.Session;
10 import org.hibernate.SessionFactory;
11
12     /**
13     *
14     * @author hp
15     */
16     public class KaryawanDaoImpl implements KaryawanDao {
17         private final SessionFactory sessionFactory;
18
19         public KaryawanDaoImpl(SessionFactory sessionFactory) {
20             this.sessionFactory = sessionFactory;
21         }
22
23         @Override
24         @SuppressWarnings("ConvertToTryWithResources")
25         public void save(Karyawan karyawan) {
26             Session session = sessionFactory.openSession();
27             try {
28                 session.beginTransaction();
29                 session.save(karyawan);
30                 session.getTransaction().commit();
31             } catch (Exception e) {
32                 session.getTransaction().rollback();
33             } finally {
34                 session.close();
35             }
36         }
37
38         @Override
39         @SuppressWarnings("ConvertToTryWithResources")
40         public void update(Karyawan karyawan) {
41             Session session = sessionFactory.openSession();
42             try {
43                 session.beginTransaction();
44                 session.update(karyawan);
45                 session.getTransaction().commit();
46             } catch (Exception e) {

```

I

```
...src KaryawanDao.java KaryawanDaoImpl.java KaryawanTableModel.java Karyawan.java KaryawanServiceImpl.java KaryawanService.java KaryawanController.java App.java
32 } finally {
46     session.getTransaction().rollback();
47 } finally {
48     session.close();
49 }
50 }
51
52 @Override
53 @SuppressWarnings("ConvertToTryWithResources")
54 public void delete(Karyawan karyawan) {
55     Session session = sessionFactory.openSession();
56     try {
57         session.beginTransaction();
58         session.delete(karyawan);
59         session.getTransaction().commit();
60     } catch (Exception e) {
61         session.getTransaction().rollback();
62     } finally {
63         session.close();
64     }
65 }
66
67 @Override
68 @SuppressWarnings("ConvertToTryWithResources")
69 public Karyawan getKaryawan(String nipKaryawan) {
70     Session session = sessionFactory.openSession();
71     try {
72         session.beginTransaction();
73         Karyawan karyawan = (Karyawan) session.get(Karyawan.class, nipKaryawan);
74         session.getTransaction().commit();
75         return karyawan;
76     } catch (Exception e) {
77         session.getTransaction().rollback();
78         return null;
79     } finally {
80         session.close();
81     }
82 }
83
84 @Override
85 @SuppressWarnings("ConvertToTryWithResources")
86 public List<Karyawan> getKaryawans() {
87     Session session = sessionFactory.openSession();
88     try {
89         session.beginTransaction();
90         List<Karyawan> karyawans = session.createCriteria(Karyawan.class).list();
91
92         session.getTransaction().commit();
93         return karyawans;
94     } catch (Exception e) {
95         session.getTransaction().rollback();
96         return null;
97     } finally {
98         session.close();
99     }
100 }
101 }
```

- JabKaryawan.java

```
...src JabKaryawan.java KaryawanDao.java KaryawanDaoImpl.java KaryawanTableModel.java Karyawan.java KaryawanServiceImpl.java KaryawanService.java KaryawanController.java App.java
Source History
1 /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6 package springHibernate.model;
7
8 /**
9  *
10  * @author hp
11  */
12 public enum JabKaryawan {
13     MANAGER, SUPERVISOR, STAFF
14 }
15
```

- Karyawan.java

```

1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6
7  package springhibernate.model;
8  import java.io.Serializable;
9  import java.util.Date;
10 import javax.persistence.Column;
11 import javax.persistence.Entity;
12 import javax.persistence.EnumType;
13 import javax.persistence.Enumerated;
14 import javax.persistence.GeneratedValue;
15 import javax.persistence.Id;
16 import javax.persistence.Table;
17 import javax.persistence.Temporal;
18 import javax.persistence.TemporalType;
19 import org.hibernate.annotations.GenericGenerator;
20
21 /**
22 *
23 * @author faiqah
24 */
25
26 @Entity
27 @Table(name = "tbl_karyawan")
28 public class Karyawan implements Serializable {
29
30     @Id
31     @Column(name = "nip_karyawan", length = 10)
32     private Integer nipKaryawan;
33     public void setNipKaryawan(Integer nipKaryawan){
34         this.nipKaryawan = nipKaryawan;
35     }
36     public Integer getNipKaryawan(){
37         return nipKaryawan;
38     }
39
40     @Column(name = "nama_karyawan", length = 50)
41     private String namaKaryawan;
42     public void setNamaKaryawan(String namaKaryawan){
43         this.namaKaryawan = namaKaryawan;
44     }
45     public String getNamaKaryawan(){
46         return namaKaryawan;
47     }
48
49     @Column(name = "alm_karyawan", length = 50)
50     private String almKaryawan;
51     public void setAlmKaryawan(String almKaryawan){
52         this.almKaryawan = almKaryawan;
53     }
54     public String getAlmKaryawan(){
55         return almKaryawan;
56     }
57
58     @Column(name = "jab_karyawan", length = 10)
59     @Enumerated(EnumType.STRING)
60     private JabKaryawan jabKaryawan;
61     public void setJabKaryawan(JabKaryawan jabKaryawan){
62         this.jabKaryawan = jabKaryawan;
63     }
64     public JabKaryawan getJabKaryawan(){
65         return jabKaryawan;
66     }
67
68     @Column(name = "tgl_lahir")
69     @Temporal(TemporalType.DATE)
70     private Date tglLahir;
71     public void setTglLahir(Date tglLahir){
72         this.tglLahir = tglLahir;
73     }
74     public Date getTglLahir(){
75         return tglLahir;
76     }
77
78     @Column(name = "tgl_kerja")
79     @Temporal(TemporalType.DATE)
80     private Date tglKerja;
81     public void setTglKerja(Date tglKerja){
82         this.tglKerja = tglKerja;
83     }
84     public Date getTglKerja(){
85         return tglKerja;
86     }
87
88     @Column(name = "gaji_karyawan")
89     private Double gajiKaryawan;
90
91

```

```

46
47     @Column(name = "alm_karyawan", length = 50)
48     private String almKaryawan;
49     public void setAlmKaryawan(String almKaryawan){
50         this.almKaryawan = almKaryawan;
51     }
52     public String getAlmKaryawan(){
53         return almKaryawan;
54     }
55
56     @Column(name = "jab_karyawan", length = 10)
57     @Enumerated(EnumType.STRING)
58     private JabKaryawan jabKaryawan;
59     public void setJabKaryawan(JabKaryawan jabKaryawan){
60         this.jabKaryawan = jabKaryawan;
61     }
62     public JabKaryawan getJabKaryawan(){
63         return jabKaryawan;
64     }
65
66     @Column(name = "tgl_lahir")
67     @Temporal(TemporalType.DATE)
68     private Date tglLahir;
69     public void setTglLahir(Date tglLahir){
70         this.tglLahir = tglLahir;
71     }
72     public Date getTglLahir(){
73         return tglLahir;
74     }
75
76     @Column(name = "tgl_kerja")
77     @Temporal(TemporalType.DATE)
78     private Date tglKerja;
79     public void setTglKerja(Date tglKerja){
80         this.tglKerja = tglKerja;
81     }
82     public Date getTglKerja(){
83         return tglKerja;
84     }
85
86     @Column(name = "gaji_karyawan")
87     private Double gajiKaryawan;
88
89
90

```

```

78  @Column(name = "tgl_kerja")
91  public void setGajiKaryawan(Double gajiKaryawan) {
92      this.gajiKaryawan = gajiKaryawan;
93  }
94  public Double getGajiKaryawan() {
95      return gajiKaryawan;
96  }
97
98  @Column(name = "bonus")
99  private Double bonus;
100 public void setBonus(Double bonus) {
101     this.bonus = bonus;
102 }
103 public Double getBonus() {
104     return bonus;
105 }
106
107 @Column(name = "pph")
108 private Double pph;
109 public void setPph(Double pph) {
110     this.pph = pph;
111 }
112 public Double getPph() {
113     return pph;
114 }
115
116 @Column(name = "total_gaji")
117 private Double totalGaji;
118 public void setTotalGaji(Double totalGaji) {
119     this.totalGaji = totalGaji;
120 }
121 public Double getTotalGaji() {
122     return totalGaji;
123 }
124
125 public void hitungGaji() {
126     switch (jabKaryawan) {
127         case MANAGER:
128             this.gajiKaryawan = 1500000.0; // Gaji Manager
129             this.bonus = this.gajiKaryawan * 0.50; // Bonus Manager 50%
130             break;
131         case SUPERVISOR:
132             this.gajiKaryawan = 1000000.0; // Gaji Supervisor
133             this.bonus = this.gajiKaryawan * 0.40; // Bonus Supervisor 40%
134             break;
135         case STAFF:
136             this.gajiKaryawan = 500000.0; // Gaji Staff
137             this.bonus = this.gajiKaryawan * 0.30; // Bonus Staff 30%
138             break;
139         default:
140             this.gajiKaryawan = 0.0; // Default if job position is not recognized
141             this.bonus = 0.0;
142     }
143     this.pph = (this.gajiKaryawan + this.bonus) * 0.05; // PPH 5%
144     this.totalGaji = this.gajiKaryawan + this.bonus - this.pph; // Total Gaji
145 }
146 }

```

- KaryawanService.java

```

...xml  JabKaryawan.java  KaryawanDao.java  KaryawanDaoImpl.java  KaryawanTableModel.java  Karyawan.java  KaryawanServiceImpl.java  KaryawanService.java  KaryawanConti
Source  History
1  /*
2   * To change this license header, choose License Headers in Project Properties.
3   * To change this template file, choose Tools | Templates
4   * and open the template in the editor.
5   */
6  package springHibernate.service;
7
8  import java.util.List;
9  import springHibernate.model.Karyawan;
10
11
12  /**
13   *
14   * @author hp
15   */
16  public interface KaryawanService {
17      public void save(Karyawan karyawan);
18      public void update(Karyawan karyawan);
19      public void delete(Integer nipKaryawan);
20      public Karyawan getKaryawan(Integer nipKaryawan);
21      public List<Karyawan> getKaryawans();
22      public List<Karyawan> getKaryawansByBulan(String bulan);
23      public List<Karyawan> getKaryawansByNipAndBulan(int nip, String bulan);
24  }
25

```


- KaryawanServiceImpl.java

```

...xm JobKaryawan.java x KaryawanDao.java x KaryawanDaoImpl.java x KaryawanTableModel.java x Karyawan.java x KaryawanServiceImpl.java x KaryawanService.java x KaryawanConti
Source History
1 package springHibernate.service;
2
3 import java.util.List;
4 import org.hibernate.SessionFactory;
5 import org.springframework.beans.factory.annotation.Autowired;
6 import org.springframework.stereotype.Service;
7 import org.springframework.transaction.annotation.Transactional;
8 import springHibernate.model.Karyawan;
9
10 @Service("KaryawanService")
11 @Transactional(readOnly = true)
12 public class KaryawanServiceImpl implements KaryawanService {
13
14     @Autowired
15     private SessionFactory sessionFactory;
16
17     @Transactional
18     @Override
19     public void save(Karyawan karyawan) {
20         sessionFactory.getCurrentSession().save(karyawan);
21     }
22
23     @Transactional
24     @Override
25     public void update(Karyawan karyawan) {
26         sessionFactory.getCurrentSession().update(karyawan);
27     }
28
29     @Transactional
30     @Override
31     public void delete(Integer nipKaryawan) {
32         Karyawan karyawan = (Karyawan) sessionFactory.getCurrentSession().get(Karyawan.class, nipKaryawan);
33         if (karyawan != null) {
34             sessionFactory.getCurrentSession().delete(karyawan);
35         }
36     }
37
38     @Override
39     public Karyawan getKaryawan(Integer nipKaryawan) {
40         return (Karyawan) sessionFactory.getCurrentSession().get(Karyawan.class, nipKaryawan);
41     }
42
43     @Override
44     public List<Karyawan> getKaryawans() {
45         return sessionFactory.getCurrentSession().createCriteria(Karyawan.class).list();
46     }
47
48     @Override
49     public List<Karyawan> getKaryawansByBulan(String bulan) {
50         String hql = "FROM Karyawan k WHERE FUNCTION('MONTH', k.tglKerja) = :bulan";
51         return sessionFactory.getCurrentSession().createQuery(hql)
52             .setParameter("bulan", getBulanNumber(bulan))
53             .list();
54     }
55
56     @Override
57     public List<Karyawan> getKaryawansByNipAndBulan(int nip, String bulan) {
58         String hql = "FROM Karyawan k WHERE k.nipKaryawan = :nip AND FUNCTION('MONTH', k.tglKerja) = :bulan";
59         return sessionFactory.getCurrentSession().createQuery(hql)
60             .setParameter("nip", nip)
61             .setParameter("bulan", getBulanNumber(bulan))
62             .list();
63     }
64
65     private int getBulanNumber(String bulan) {
66         switch (bulan) {
67             case "Januari": return 1;
68             case "Februari": return 2;
69             case "Maret": return 3;
70             case "April": return 4;
71             case "Mei": return 5;
72             case "Juni": return 6;
73             case "Juli": return 7;
74             case "Agustus": return 8;
75             case "September": return 9;
76             case "Oktober": return 10;
77             case "November": return 11;
78             case "Desember": return 12;
79             default: return 0;
80         }
81     }
82
83 }

```

- Karyawan View.java

```

1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6
7  package springHibernate.view;
8
9  import com.mysql.jdbc.Connection;
10 import java.io.File;
11 import java.util.HashMap;
12 import java.util.Map;
13 import javax.swing.JOptionPane;
14 import springHibernate.controller.KaryawanController;
15 import javax.swing.JTable;
16 import javax.swing.JTextField;
17 import net.sf.jasperreports.engine.JasperCompileManager;
18 import net.sf.jasperreports.engine.JasperFillManager;
19 import net.sf.jasperreports.engine.JasperPrint;
20 import net.sf.jasperreports.engine.JasperReport;
21 import net.sf.jasperreports.engine.design.JasperDesign;
22 import net.sf.jasperreports.engine.xml.JRXmlLoader;
23 import net.sf.jasperreports.view.JasperViewer;
24 import springHibernate.koneksi;
25
26 /**
27  *
28  * @author hp
29  */
30 public class KaryawanView extends javax.swing.JFrame {
31     private final KaryawanController karyawanController = new KaryawanController(this);
32
33     Connection c;
34     JasperReport JasRep;
35     JasperPrint JasPri;
36     JasperDesign JasDes;
37     Map param = new HashMap();
38
39     /**
40      * Creates new form KaryawanView
41      */
42     public KaryawanView() {
43         initComponents();
44         karyawanController.tampilData();
45         c = (Connection) koneksi.koneksi();
46     }
47
48     /**

```

```

258 private void deleteActionPerformed(java.awt.event.ActionEvent evt) {
259     karyawanController.deleteKaryawan();
260 }
261
262 private void simpanActionPerformed(java.awt.event.ActionEvent evt) {
263     karyawanController.saveKaryawan();
264 }
265
266 private void updateActionPerformed(java.awt.event.ActionEvent evt) {
267     karyawanController.updateKaryawan();
268 }
269
270 private void tabelMouseClicked(java.awt.event.MouseEvent evt) {
271     karyawanController.show();
272 }
273
274 private void printActionPerformed(java.awt.event.ActionEvent evt) {
275     try {
276         File report = new File ("src/springHibernate/reportKaryawan.jrxml");
277         JasDes = JRXmlLoader.load(report);
278         param.clear();
279         JasRep = JasperCompileManager.compileReport(JasDes);
280         JasPri = JasperFillManager.fillReport(JasRep, param, c);
281         JasperViewer.viewReport(JasPri, false);
282     } catch (Exception e) {
283         JOptionPane.showMessageDialog(null, e);
284     }
285 }
286
287 private void jabKaryawanActionPerformed(java.awt.event.ActionEvent evt) {
288     // TODO add your handling code here:
289 }

```

```

290 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
291     // Hide the current window
292     this.setVisible(false);
293
294     // Create and display the SlipGajiView
295     SlipGajiView slipGajiView = new SlipGajiView();
296     slipGajiView.setVisible(true);
297 }
298
299
300 public JTextField getNipKaryawan() {
301     return nipKaryawan;
302 }
303
304 public JTextField getNamaKaryawan() {
305     return namaKaryawan;
306 }
307
308 public JTextField getAlmKaryawan() {
309     return almKaryawan;
310 }
311
312 public JTextField getTglLahir() {
313     return tglLahir;
314 }
315
316 public JTextField getJabKaryawan() {
317     return jabKaryawan;
318 }
319
320 public JTextField getTglKerja() {
321     return tglKerja;
322 }
323
324 public JTable getTabel() {
325     return tabel;
326 }

```

```

...ave KaryawanDaoImpl.java x KaryawanTableModel.java x Karyawan.java x KaryawanServiceImpl.java x KaryawanService.java x KaryawanController.java x App.java x koneksi.java x
320 }
349 //</editor-fold>
350 //</editor-fold>
351 //</editor-fold>
352 //</editor-fold>
353 //</editor-fold>
354
355 /* Create and display the form */
356 java.awt.EventQueue.invokeLater(new Runnable() {
357     public void run() {
358         new KaryawanView().setVisible(true);
359     }
360 });
361
362
363 // Variables declaration - do not modify
364 private javax.swing.JTextField almKaryawan;
365 private javax.swing.JButton delete;
366 private javax.swing.JButton jButton1;
367 private javax.swing.JLabel jLabel1;
368 private javax.swing.JLabel jLabel2;
369 private javax.swing.JLabel jLabel3;
370 private javax.swing.JLabel jLabel4;
371 private javax.swing.JLabel jLabel5;
372 private javax.swing.JLabel jLabel6;
373 private javax.swing.JLabel jLabel7;
374 private javax.swing.JScrollPane jScrollPane1;
375 private javax.swing.JTextField jabKaryawan;
376 private javax.swing.JTextField namaKaryawan;
377 private javax.swing.JTextField nipKaryawan;
378 private javax.swing.JButton print;
379 private javax.swing.JButton simpan;

```

- SlipGajiView.java

```

...ave KaryawanTableModel.java x Karyawan.java x KaryawanServiceImpl.java x KaryawanService.java x KaryawanController.java x App.java x koneksi.java x KaryawanView.java x SI
Source Design History
1 /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6 package springHibernate.view;
7
8 import javax.swing.*;
9 import java.awt.*;
10 import java.awt.event.ActionEvent;
11 import java.awt.event.ActionListener;
12 import java.util.List;
13 import springHibernate.model.Karyawan;
14 import springHibernate.configuration.KaryawanTableModel;
15 import springHibernate.App;
16
17 /**
18  *
19  * @author hp
20  */
21 public class SlipGajiView extends javax.swing.JFrame {
22
23     /**
24      * Creates new form SlipGajiView
25      */
26     public SlipGajiView() {
27         initComponents();
28     }
29
30     /**
31      * This method is called from within the constructor to initialize the form.
32      * WARNING: Do NOT modify this code. The content of this method is always
33      * regenerated by the Form Editor.
34      */
35     @SuppressWarnings("unchecked")
36     Generated Code

```

```

...ave KaryawanTableModel.java x Karyawan.java x KaryawanServiceImpl.java x KaryawanService.java x KaryawanController.java x App.java x koneksi.java x KaryawanView.java x SI
Source Design History
156 private void bulanActionPerformed(java.awt.event.ActionEvent evt) {
157     loadKaryawanData();
158 }
159
160 private void searchActionPerformed(java.awt.event.ActionEvent evt) {
161     loadKaryawanData();
162 }
163
164 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
165     // Hide the current window
166     this.setVisible(false);
167
168     // Create and display the SlipGajiView
169     KaryawanView karyawanView = new KaryawanView();
170     karyawanView.setVisible(true);
171 }
172
173 /**
174  * @param args the command line arguments
175  */
176 public static void main(String args[]) {
177     ...46 lines
178 }
179
180 private void loadKaryawanData() {
181     String selectedBulan = (String) bulan.getSelectedItem();
182     String nipText = nipKaryawan.getText();
183
184     if (nipText != null && !nipText.trim().isEmpty()) {
185         try {
186             int nip = Integer.parseInt(nipText);
187             List<Karyawan> karyawans = App.getKaryawanService().getKaryawansByNipAndBulan(nip, selectedBulan);
188             KaryawanTableModel karyawanTableModel = new KaryawanTableModel(karyawans);
189             jTable1.setModel(karyawanTableModel);
190         } catch (NumberFormatException ex) {
191             JOptionPane.showMessageDialog(this, "NIP harus berupa angka", "Error", JOptionPane.ERROR_MESSAGE);
192         }
193     } else {
194         List<Karyawan> karyawans = App.getKaryawanService().getKaryawansByBulan(selectedBulan);
195         KaryawanTableModel karyawanTableModel = new KaryawanTableModel(karyawans);
196         jTable1.setModel(karyawanTableModel);
197     }
198 }
199
200 }
201
202 }

```

Output.

Sistem Pengolahan Data Karyawan

NIP

50420444

Simpan

Nama Karyawan

Udin

Ubah

Alamat

Jogja

Hapus

Tanggal Lahir

2001-11-07

Jabatan

STAFF

Tanggal Masuk Kerja

2024-01-05

NIP	Nama Karya...	Alamat	Jabatan	Tanggal Lahir	Tanggal Mula...	Gaji Pokok	Bonus	PPH	Total Gaji
50420440	Faiqah	Depok	STAFF	2002-11-27	2024-05-17	500000.0	150000.0	32500.0	617500.0
50420441	Farras	Jakarta	STAFF	2002-03-02	2024-05-16	500000.0	150000.0	32500.0	617500.0
50420442	Fitrianthasa	Bandung	SUPERVISOR	2002-04-25	2024-05-18	1000000.0	400000.0	70000.0	1330000.0
50420443	Fahrah	Depok	STAFF	2002-07-01	2024-03-01	500000.0	150000.0	32500.0	617500.0

Slip Gaji

Cetak Laporan Gaji

Sistem Pengolahan Data Karyawan

NIP

50420444

Simpan

Nama Karyawan

Udin

Ubah

Alamat

Jogja

Hapus

Tanggal Lahir

2001-11-07

Jabatan

STAFF

Tanggal Masuk Kerja

2024-01-05

NIP	Nama Karya...	Alamat	Jabatan	Ta	Bonus	PPH	Total Gaji		
50420440	Faiqah	Depok	STAFF	200	000.0	32500.0	617500.0		
50420441	Farras	Jakarta	STAFF	200	000.0	32500.0	617500.0		
50420442	Fitrianthasa	Bandung	SUPERVISOR	200	000.0	70000.0	1330000.0		
50420443	Fahrah	Depok	STAFF	2002-07-01	2024-03-01	500000.0	150000.0	32500.0	617500.0

Slip Gaji

Cetak Laporan Gaji

Info

Data Berhasil Disimpan

OK

Sistem Pengolahan Data Karyawan

NIP

Simpan

Nama Karyawan

Ubah

Alamat

Hapus

Tanggal Lahir

Jabatan

Tanggal Masuk Kerja

NIP	Nama Karya...	Alamat	Jabatan	Tanggal Lahir	Tanggal Mula...	Gaji Pokok	Bonus	PPH	Total Gaji
50420440	Faiqah	Depok	STAFF	2002-11-27	2024-05-17	500000.0	150000.0	32500.0	617500.0
50420441	Farras	Jakarta	STAFF	2002-03-02	2024-05-16	500000.0	150000.0	32500.0	617500.0
50420442	Filrianthasa	Bandung	SUPERVISOR	2002-04-25	2024-05-18	1000000.0	400000.0	70000.0	1330000.0
50420443	Fahrah	Depok	STAFF	2002-07-01	2024-03-01	500000.0	150000.0	32500.0	617500.0
50420444	Udin	Jogja	STAFF	2001-11-07	2024-01-05	500000.0	150000.0	32500.0	617500.0

Slip Gaji

Cetak Laporan Gaji

Kembali

CETAK SLIP GAJI


50420440

Cari

Mei

NIP	Nama Ka...	Alamat	Jabatan	Tanggal L...	Tanggal ...	Gaji Pokok	Bonus	PPH	Total Gaji
50420440	Faiqah	Depok	STAFF	2002-11-...	2024-05-...	500000.0	150000.0	32500.0	617500.0

Cetak Slip Gaji



Kembali

CETAK SLIP GAJI


50420441

Cari

Mei

NIP	Nama Ka	Alamat	Jabatan	Tanggal L...	Tanggal ...	Gaji Pokok	Bonus	PPH	Total Gaji
50420441	Farras	Jakarta	STAFF	2002-03-...	2024-05-...	500000.0	150000.0	32500.0	617500.0

Cetak Slip Gaji



Kembali

CETAK SLIP GAJI

50420442

Cari

Mei

NIP	Nama Ka	Alamat	Jabatan	Tanggal L...	Tanggal ...	Gaji Pokok	Bonus	PPH	Total Gaji
50420442	Fitriantha...	Bandung	SUPERVI...	2002-04-...	2024-05-...	1000000.0	400000.0	70000.0	1330000.0

Cetak Slip Gaji

Kembali

CETAK SLIP GAJI

50420443

Cari

Maret

NIP	Nama Ka...	Alamat	Jabatan	Tanggal L...	Tanggal ...	Gaji Pokok	Bonus	PPH	Total Gaji
50420443	Fahrah	Depok	STAFF	2002-07-...	2024-03-...	500000.0	150000.0	32500.0	617500.0

Cetak Slip Gaji

Kembali

CETAK SLIP GAJI

50420444

Cari

Januari

NIP	Nama Ka...	Alamat	Jabatan	Tanggal L...	Tanggal ...	Gaji Pokok	Bonus	PPH	Total Gaji
50420444	Udin	Jogja	STAFF	2001-11-...	2024-01-...	500000.0	150000.0	32500.0	617500.0

Cetak Slip Gaji