



ISPARTA UYGULAMALI BİLİMLER ÜNİVERSİTESİ

TEKNOLOJİ FAKÜLTESİ

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

NESNE TABANLI PROGRAMLAMA DERSİ

Dersin Öğretim Üyesi

Dr. Öğr. Üyesi Cevriye ALTINTAŞ

Dersin Öğretim Elemanı

Arş. Gör. Ahmet Bestami KÖSE

Öğrenci Ad Soyadı

BAYRAM BAYRAKTAR // HÜSEYİN KARABULUT

Öğrenci Numarası

2112721011 // 2112721044

BEŞİKTAŞ BANK A.Ş OTOMASYONU



Görsel 1

```
* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
* Click nbfs://nbhost/SystemFileSystem/Templates/GuiForms/JFrame.java to edit this template
*/
package banka;

public class Login extends javax.swing.JFrame {

    /**
     * Creates new form giris
     */
    public Login() {
        initComponents();
    }

    /**
     * This method is called from within the constructor to initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    Generated Code

    private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        new MusteriLogin().show();
        dispose();
    }

    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        new MudurLogin().show();
        dispose();
    }

    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
    }
}
```

Görsel 1’de yaptığımız otomasyonun giriş ekranı. Bu ekranda 3 farklı kullanıcı girişi bulunmaktadır.

Bu kullanıcılar Müdür, Bankacı ve Müşteri. Otomasyona girişi sağlamak için kullanıcılar kendi bilgileri ile otomasyona giriş yapmalıdırlar.

Otomasyonu müdür kullanıcısı ile tanımaya başlayalım;

Müdür, veri tabanında önceden kayıt edilmeli ve işlemler diğer şekilde devam etmelidir.



```
/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
Generated Code

private void txtmudursifreActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String username = txtmudurtc.getText();
    String password = txtmudursifre.getText();
    String sqlquery = "SELECT * FROM bankmanager WHERE bankmanagerTc='"+username+"' AND bankmanagerPass='"+password+"'";
    try{
        Connection con = conn.connDb();
        PreparedStatement pst = con.prepareStatement( string:sqlquery);
        ResultSet rs = pst.executeQuery();
        if(!rs.next()){
            JOptionPane.showMessageDialog( parentComponent:null, message: "TC veya Şifre Hatalı");
        }else{
            JOptionPane.showMessageDialog( parentComponent:null, message: "Giriş Başarılı");
            new MudurView().show();
            dispose();
        }
    }catch(Exception e){
        JOptionPane.showMessageDialog( parentComponent:null, message:e );
    }
}
```

Müdür kendi kişisel bilgileri ile otomasyona giriş sağlamakta. Kullanıcı otomasyona doğru bilgi girdiğinde “Giriş Başarılı” ,yanlış girildiğinde ise “TC veya Şifre Hatalı” bilgilendirme mesajları gelmektedir.



```

        tablo_update();
        txttcduzen.setText( t: "" );
        txtsifreduzen.setText( t: "" );
        txttcduzen.requestFocus();
        con.close();

    } catch (Exception e) {
        JOptionPane.showMessageDialog( parentComponent: null, message: e );
    }
}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    int row = jTable1.getSelectedRow();
    String id = jTable1.getValueAt(row, column:0).toString();
    String sqlquery = "DELETE FROM banker WHERE bankerTc='" + id + "'";

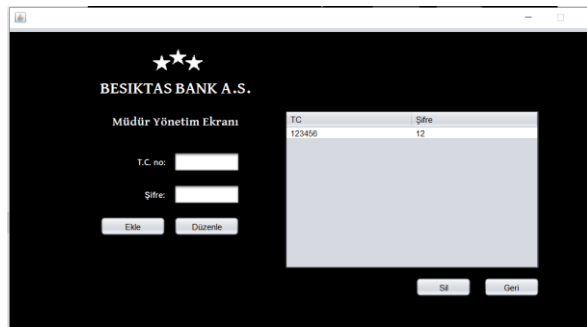
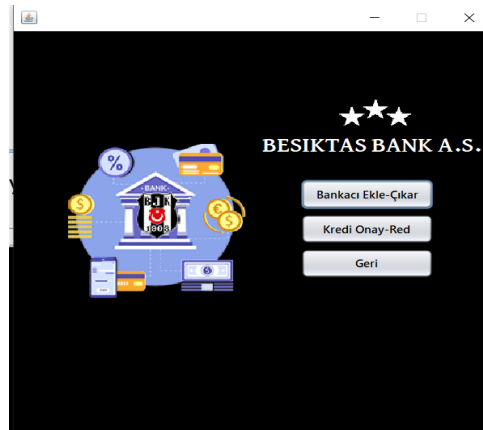
    try {
        Connection con = conn.connDb();
        Statement stm = con.createStatement();
        PreparedStatement pst = con.prepareStatement( string:sqlquery);
        stm.executeUpdate( string:sqlquery);
        JOptionPane.showMessageDialog( parentComponent: null, message: "Kayıt Silindi");
        tablo_update();
        txttcduzen.setText( t: "" );
        txtsifreduzen.setText( t: "" );
        txttcduzen.requestFocus();
        con.close();

    } catch (Exception e) {
        JOptionPane.showMessageDialog( parentComponent: null, message: e );
    }
}

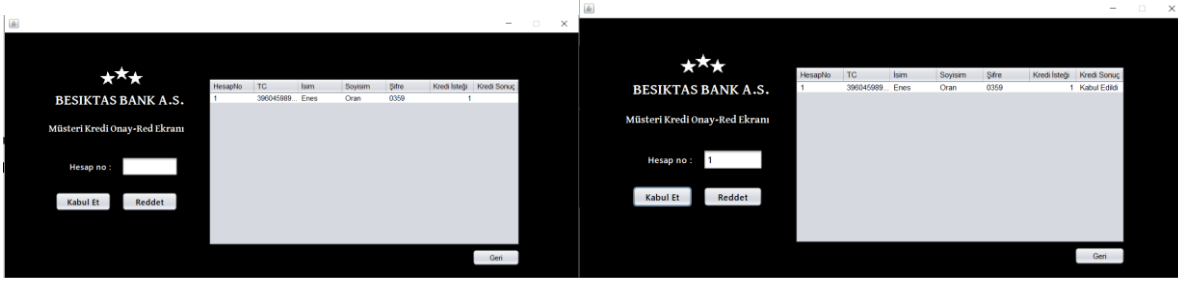
private void jButton7ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:

```

Bir müdür kullanıcısı bankacı kullanıcısını ekleme/çıkarma ve müşteri kredi onay/red işlemlerini yerine getirebilme işlevlerini yapmaktadır.



Müdür “Bankacı Ekle-Çıkar” butonuna bastığı takdirde yukarıdaki görsel açılacak. Görselde de görüldüğü üzere bankacı ekle çıkarma işlevlerini yerine getirmektedir.



İlk görselde yapılmış olan kredi başvurusu bulunmakta. Müdür sayfasına düşen kredi başvurusu müdür tarafından kabul veya red alan kredi sonucu “Kredi Sonuç” sütununda bulunmaktadır. İkinci görselde bunun örneği bulunmakta. Müdürün yaptığı işlevleri şimdilik bu kadar.

```

if (!rs.next()) {
    JOptionPane.showMessageDialog( parentComponent: null, message: "HesapNo Hatalı");
} else {
    String sqlquery2 = "UPDATE customeraccount SET customerCreditResult='" + a + "'WHERE accountNo='" + username + "'";
    PreparedStatement pst1 = con.prepareStatement( string:sqlquery2);
    pst1.execute();

    JOptionPane.showMessageDialog( parentComponent: null, message: "Kabul Edildi");

}
con.close();
} catch (Exception e) {
    JOptionPane.showMessageDialog( parentComponent: null, message: e);
}
tablo_update();
}

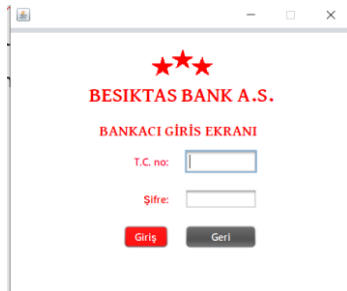
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String username = txthesapno.getText();
    String a = "Red Edildi";
    String sqlquery = "SELECT * FROM customeraccount WHERE accountNo='" + username + "'";
    try {
        Connection con = conn.connDb();
        PreparedStatement pst = con.prepareStatement( string:sqlquery);

        ResultSet rs = pst.executeQuery();
        if (!rs.next()) {
            JOptionPane.showMessageDialog( parentComponent: null, message: "HesapNo Hatalı");
        } else {
            String sqlquery2 = "UPDATE customeraccount SET customerCreditResult='" + a + "'WHERE accountNo='" + username + "'";
            PreparedStatement pst1 = con.prepareStatement( string:sqlquery2);
            pst1.execute();

            JOptionPane.showMessageDialog( parentComponent: null, message: "Red Edildi");
        }
    }
}

```

Şimdi de sıra Bankacıya geldi tanıtmaya başlayalım;



Bankacı kendi bilgileri ile girişi sağladıktan sonra yetkili olduğu alan açılıp verilen yetkiler doğrultusunda işlemler yapabilecek.

```
// TODO add your handling code here:
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String username = txtbankacitc.getText();
    String password = txtbankacisifre.getText();
    String sqlquery = "SELECT * FROM banker WHERE bankerTc='" + username + "' AND bankerPass='" + password + "'";
    try {
        Connection con = conn.connDb();
        PreparedStatement pst = con.prepareStatement(string:sqlquery);
        ResultSet rs = pst.executeQuery();
        if (!rs.next()) {
            JOptionPane.showMessageDialog(parentComponent:null, message:"TC veya Şifre Hatalı");
        } else {
            JOptionPane.showMessageDialog(parentComponent:null, message:"Giriş Başarılı");
            new BankaciView().show();
            dispose();
        }
    } catch (Exception e) {
        JOptionPane.showMessageDialog(parentComponent:null, message:e);
    }
}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    new Login().show();
    dispose();
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    Look and feel setting code (optional)
}
```



Bankacı Ekran bazı buton kodları.

```
jLabel13 = new javax.swing.JLabel();
jLabel14 = new javax.swing.JLabel();
jLabel15 = new javax.swing.JLabel();

setDefaultCloseOperation(operation: javax.swing.WindowConstants.EXIT_ON_CLOSE);
setResizable(resizable: false);

jPanel1.setBackground(new java.awt.Color(r:255, g:255, b:255));

jButton1.setFont(new java.awt.Font(name:"Yu Gothic UI Semibold", style:0, size:12)); // NOI18N
jButton1.setText(text:"Müşteri Ekleme-Çıkarma");
jButton1.setToolTipText(text:"");
jButton1.setCursor(new java.awt.Cursor(type: java.awt.Cursor.HAND_CURSOR));
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

jButton2.setFont(new java.awt.Font(name:"Yu Gothic UI Semibold", style:0, size:12)); // NOI18N
jButton2.setText(text:"Para Yatır-Çek");
jButton2.setCursor(new java.awt.Cursor(type: java.awt.Cursor.HAND_CURSOR));
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

jButton3.setFont(new java.awt.Font(name:"Yu Gothic UI Semibold", style:0, size:12)); // NOI18N
jButton3.setText(text:"Banka Hesabı Aç");
jButton3.setCursor(new java.awt.Cursor(type: java.awt.Cursor.HAND_CURSOR));
jButton3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton3ActionPerformed(evt);
    }
});
```

Bir bankacı banka hesabı açma, para yatır-çek ve müşteri ekleme çıkarma işlemleri yapabilmektedir.

BESIKTAS BANK A.S.

Hesap İşlemleri

T.C. no:

İsim:

Soyisim:

Şifre:

HesapNo	TC	İsim	Soyisim	Şifre	Bakiye
1	3960459...	Enes	Oran	0359	0

Girilen bilgiler doğrultusunda yan ekranda da görüldüğü gibi müşteriye banka hesabı açabilmekte, aynı zamanda müşteri silebilmekte de.

Müşteri banka hesabına para ekleme yada çıkarma işlemleri için “Para Yatır-Çek” butonuna basıp gerekli işlemleri yapabilmektedir.

Hesap işlemleri buton kodları.

```
@SuppressWarnings("unchecked")
Generated Code

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String tcduzen = txttcduzen.getText();
    String isimduzen = txtisimduzen.getText();
    String soyisimduzen = txtsoyisimduzen.getText();
    String sifreduzen = txtsifreduzen.getText();
    String sqlquery = "INSERT INTO customeraccount(customerTc,customerName,customerSurname,customerPass,customerBalance) VALUES('" + tcduzen + "','" +
    if (txttcduzen.getText().equals("")) {
        JOptionPane.showMessageDialog(parentComponent: null, message: "Lütfen TC Girin");
    }
    if (txtisimduzen.getText().equals("")) {
        JOptionPane.showMessageDialog(parentComponent: null, message: "Lütfen İsim Girin");
    }
    if (txtsoyisimduzen.getText().equals("")) {
        JOptionPane.showMessageDialog(parentComponent: null, message: "Lütfen Soyisim Girin");
    } else if (txtsifreduzen.getText().equals("")) {
        JOptionPane.showMessageDialog(parentComponent: null, message: "Lütfen Şifre Girin");
    }
    try {
        Connection con = conn.connDb();
        Statement stm = con.createStatement();
        PreparedStatement pst = con.prepareStatement(sqlquery);

        stm.executeUpdate(sqlquery);
        JOptionPane.showMessageDialog(parentComponent: null, message: "Kayıt Edildi");
        txttcduzen.setText("");
        txtisimduzen.setText("");
        txtsoyisimduzen.setText("");
        txtsifreduzen.setText("");
        txttcduzen.requestFocus();
        tablo_update();
        con.close();
    }
}
```



BESIKTAS BANK A.S.

Müşteri hesap no:

Para yatırma:

Para Çekme:

accountNo	customerTc	customerN	customerS	customerP	customerB	customerC	customerC
1	39604598958	Enes	Oran	0359	1250	1	Kabul Edildi
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Hesap numarası 1 olan kişi hesabına para yatırma işlemi gerçekleşti. Veri tabanında da doğruluğu kanıtlanmış oldu. Şimdi de para çekme işlemleri için aynısı yapalım.



BESIKTAS BANK A.S.

Müşteri hesap no:

Para yatırma:

Para Çekme:

accountNo	customerTc	customerN	customerS	customerP	customerB	customerC	customerC
1	39604598958	Enes	Oran	0359	800	1	Kabul Edildi
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Hesap numarası 1 olan kişi hesabına para çekme işlemi gerçekleşti. Veri tabanında da doğruluğu kanıtlanmış oldu.

BESIKTAS BANK A.S.

Müşteri İşlemleri

T.C.no:

İsim:

Soyisim:

Şifre:

TC	İsim	Soyisim	Şifre
1903	Ricardo	Quaresma	1903

BESIKTAS BANK A.S.

Müşteri İşlemleri

T.C.no:

İsim:

Soyisim:

Şifre:

TC	İsim	Soyisim	Şifre
----	------	---------	-------

Burada ise bankacı herhangi bir müşteri bilgilerini girip müşteri ekleme çıkarma yapmaktadır.

Müşteri İşlemleri buton kodları.

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    String username = txtmusterihesapno.getText();  
    String miktar1 = txtyatirma.getText();  
    int intmiktar1 = Integer.parseInt(s:miktar1);  
    String sqlquery = "SELECT * FROM customeraccount WHERE accountNo='" + username + "'";  
    try {  
        Connection con = conn.connDb();  
        PreparedStatement pst = con.prepareStatement(string:sqlquery);  
  
        ResultSet rs = pst.executeQuery();  
        if (!rs.next()) {  
            JOptionPane.showMessageDialog(parentComponent:null, message:"HesapNo Hatalı");  
        } else {  
            String bakiye = rs.getString(string:"customerBalance");  
            int toplambakiye = Integer.parseInt(s:bakiye) + intmiktar1;  
            String sqlquery2 = "UPDATE customeraccount SET customerBalance='" + toplambakiye + "'WHERE accountNo='" + username + "'";  
            PreparedStatement pst1 = con.prepareStatement(string:sqlquery2);  
            pst1.execute();  
  
            JOptionPane.showMessageDialog(parentComponent:null, message:"Para Yatırıldı");  
        }  
        con.close();  
    } catch (Exception e) {  
        JOptionPane.showMessageDialog(parentComponent:null, message:e);  
    }  
}  
  
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    String username = txtmusterihesapno.getText();  
    String miktar1 = txtcekme.getText();  
    int intmiktar1 = Integer.parseInt(s:miktar1);  
    String sqlquery = "SELECT * FROM customeraccount WHERE accountNo='" + username + "'";
```


Müşteri kendisi için açmış olduğu banka hesabından kendisine verilen giriş bilgileri doğrultusunda hesaba giriş sağlamaktadır.

Müşteri giriş ekranı.

Giriş sağlayan müşteri doğru bilgileri girdi.

Müşteri Giriş ve onay/red bilgilendirme mesajı buton kodları.

```
[
    * Creates new form mudurgiris
    */
    public MusteriLogin() {
        initComponents();
        MusterihesapView mhg = new MusterihesapView();
    }

    /**
     * This method is called from within the constructor to initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    Generated Code

    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        String username = txtmusterito.getText();
        String password = txtmusterisifre.getText();
        String sqlquery = "SELECT * FROM customer WHERE customerTc='" + username + "' AND customerPass='" + password + "'";
        try {
            Connection con = conn.connDb();
            PreparedStatement pst = con.prepareStatement(string:sqlquery);

            ResultSet rs = pst.executeQuery();
            if (!rs.next()) {
                JOptionPane.showMessageDialog( parentComponent: null, message: "TC veya Şifre Hatalı");
            } else {
                JOptionPane.showMessageDialog( parentComponent: null, message: "Giriş Başarılı");

                new MusteriView().show();
                dispose();
            }
        } catch (Exception e) {
            JOptionPane.showMessageDialog( parentComponent: null, message: e);
        }
    }
}
```



Giriş yapan müşteri var olan hesaplarını görüntüleyip bakiye sorgulama,
Para yatır-çek ve kredi başvurusu yapabilir.
Müşteri ekranından olan buton kodları.

```
* regenerated by the Form Editor.
*/
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jPanel1 = new javax.swing.JPanel();
    jButton1 = new javax.swing.JButton();
    jButton2 = new javax.swing.JButton();
    jButton4 = new javax.swing.JButton();
    jButton7 = new javax.swing.JButton();
    jLabel1 = new javax.swing.JLabel();
    jLabel4 = new javax.swing.JLabel();
    jLabel5 = new javax.swing.JLabel();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
    setResizable(false);

    jPanel1.setBackground(new java.awt.Color(255, 0, 0));

    jButton1.setFont(new java.awt.Font("Yu Gothic UI Semibold", style:0, size:12)); // NOI18N
    jButton1.setText("Hesap Görüntüle-Bakiye Sorgula");
    jButton1.setCursor(new java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
    jButton1.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton1ActionPerformed(evt);
        }
    });

    jButton2.setFont(new java.awt.Font("Yu Gothic UI Semibold", style:0, size:12)); // NOI18N
    jButton2.setText("Para Yatır-Çek");
    jButton2.setCursor(new java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
    jButton2.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton2ActionPerformed(evt);
        }
    });
}
```



Var olan kullanıcının hesap özeti.

Kod ekran görüntüsünden bir kesit.

```
// TODO add your handling code here:
new MusteriView().show();
dispose();
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
     * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
     */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {
        java.util.logging.Logger.getLogger(MusterihesapView.class.getName()).log(level: java.util.logging.Level.SEVERE, msg: null, thrown: ex);
    } catch (InstantiationException ex) {
        java.util.logging.Logger.getLogger(MusterihesapView.class.getName()).log(level: java.util.logging.Level.SEVERE, msg: null, thrown: ex);
    } catch (IllegalAccessException ex) {
        java.util.logging.Logger.getLogger(MusterihesapView.class.getName()).log(level: java.util.logging.Level.SEVERE, msg: null, thrown: ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {
        java.util.logging.Logger.getLogger(MusterihesapView.class.getName()).log(level: java.util.logging.Level.SEVERE, msg: null, thrown: ex);
    }
    //</editor-fold>
    //</editor-fold>
    //</editor-fold>
    //</editor-fold>
    //</editor-fold>
    //</editor-fold>
}
```



Para yatırma işlemleri.

Para Yatırma butonu kodları.

```
// TODO add your handling code here:
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String username = txtmusterihesapno.getText();
    String miktar1 = txtyatirma.getText();
    int intmiktar1 = Integer.parseInt(s:miktar1);
    String sqlquery = "SELECT * FROM customeraccount WHERE accountNo='" + username + "'";
    try {
        Connection con = conn.connDb();
        PreparedStatement pst = con.prepareStatement(string:sqlquery);

        ResultSet rs = pst.executeQuery();
        if (!rs.next()) {
            JOptionPane.showMessageDialog(parentComponent:null, message:"HesapNo Hatalı");
        } else {
            String bakiye = rs.getString(string:"customerBalance");
            int toplambakiye = Integer.parseInt(s:bakiye) + intmiktar1;
            String sqlquery2 = "UPDATE customeraccount SET customerBalance='" + toplambakiye + "'WHERE accountNo='" + username + "'";
            PreparedStatement pst1 = con.prepareStatement(string:sqlquery2);
            pst1.execute();

            JOptionPane.showMessageDialog(parentComponent:null, message:"Para Yatırıldı");
        }
        con.close();
    } catch (Exception e) {
        JOptionPane.showMessageDialog(parentComponent:null, message:e);
    }
}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String username = txtmusterihesapno.getText();
    String miktar1 = txtcekme.getText();
    int intmiktar1 = Integer.parseInt(s:miktar1);
```

BESIKTAS BANK A.S.

Hesap No: 2

TC No: 1903

Kredi Başvurusu Yap

Sonuç Göster

Hesap No	TC	Kredi Durum
2	1903	Kabul Edildi

Geri

Burada ise de yapmış olduğu kredi sonuç bilgileri görüntüleme işlemlerini yapmakta.

Kredi Başvuru Yap buton kodları.

```
    } catch (Exception e) {
        JOptionPane.showMessageDialog( parentComponent:null, message:e );
    }
}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String username = txttc.getText();
    String username1 = txthesap.getText();
    String sqlquery = "SELECT * FROM customeraccount WHERE accountNo='" + username1 + "' ";

    try {
        Connection con = conn.connDb();
        PreparedStatement pst = con.prepareStatement( string:sqlquery);

        ResultSet rs = pst.executeQuery();
        if (rs.next()) {
            String sqlquery2 = "UPDATE customeraccount SET customerCreditStatus = (case customerCreditStatus "
                                + "when '1' then '0'"
                                + "end)WHERE accountNo='" + txthesap.getText() + "' ";
            PreparedStatement pst1 = con.prepareStatement( string:sqlquery2);
            pst1.execute();
            JOptionPane.showMessageDialog( parentComponent: null, message: "Başvuru Yapıldı");
        } else {
            JOptionPane.showMessageDialog( parentComponent: null, message: "Hesap No Hatalı");
        }
    } catch (Exception e) {
        JOptionPane.showMessageDialog( parentComponent: null, message: e);
    }
}

private void txthesapActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
```