

JAVA PROGRAMLAMA

Su Animsatıcı

Buse Yazgan

Yenercan Barker

İÇİNDEKİLER:

- 1. Giriş
- 2. Kod Blokları
 - a.Veritabanı Bağlantısı ve İşlemleri
 - b.Su Hatırlatıcısı Sınıfı
 - c.Su Veri Paneli
 - d.Su Kayıt Paneli
- 3. Tasarım

1.GİRİŞ

- Kullanıcının yıl içerisinde, içtiği su miktarını görüntülediği ve gün içerisinde içmesi gereken miktarı belirleyip, içtiği su miktarını girebildiği bir program.
- Hatırlatıcı sayesinde su içme durumunun bir alışkanlık haline
- gelmesine yardımcı olmaktadır.

2.Kod Blokları

a.Veritabanı Bağlantısı ve İşlemleri

```
*/
public class SqlClass {
    java.util.Date date;
    java.sql.Date sqlDate;
    waterreminder wr=new waterreminder();

    public void Ekle(){
        try {
            Class.forName("com.mysql.jdbc.Driver");
        }
        catch(ClassNotFoundException e){
            JOptionPane.showMessageDialog(null,"Hata:" + e.getMessage());
        }
        Connection baglanti=null;
        try{
            baglanti=DriverManager.getConnection("jdbc:mysql://localhost:3306/waterreminder","root","1999");
        }
        catch (SQLException e) {
            JOptionPane.showMessageDialog(null,"Hata:" + e.getMessage());
        }
    }
}
```

```
try{
    java.util.Date d = new java.util.Date();
    java.sql.Date sd = new java.sql.Date(d.getTime());
    PreparedStatement uygula=baglanti.prepareStatement("INSERT INTO reminder(Tarih,IcilenMiktari,Hedef) VALUES(?,?,?)");
    wr.Tarih=sd;
    wr.IcilenMiktar=Integer.parseInt(lblIcilenMiktar.getText());
    wr.Hedef=Integer.parseInt(lblHedef.getText());
    uygula.setDate(1, wr.Tarih);
    uygula.setInt(2, wr.IcilenMiktar);
    uygula.setInt(3, wr.Hedef);
    int donut=uygula.executeUpdate();
}
catch(SQLException e){
    JOptionPane.showMessageDialog(null,"Hata:"+e.getMessage());
}
}
```

Sql sorgularını yazmak için sınıf oluşturduk.Sınıfın içine ekleme , silme, güncelleme işlemleri için fonksiyon ekledik. Veri tabanı bağlantısını sağladıktan sonra su kayıtlarını, veritabanına eklemek için sql sorgusu yazdık, aynı zamanda silme ve güncelleme işlemleri için de aynısını yaptık.

b.Su hatırlatıcısı sınıfı

```
public class waterreminder {  
    public int waterID;  
    public Date Tarih;  
    public int IcilenMiktar;  
    public int Hedef;  
  
    public void Tarih(Date tarih){  
        Tarih=tarih;  
    }  
    public Date getTarih(){  
        return Tarih;  
    }  
    public void IcilenMiktar(int miktar){  
        IcilenMiktar=miktar;  
    }  
    public int getMiktar(){  
        return IcilenMiktar;  
    }  
    public void Hedef(int hedef ){  
        Hedef=hedef;  
    }  
    public int getHedef(){  
        return Hedef;  
    }  
}
```

Comboboxta seçilen miktara göre, artı ve eksi butonlarına basarak miktarı 150 ml arttırıp azaltıyoruz. Eğer miktar girilmemişse, bize uyarı veriyor.

c.Su Veri Paneli

```
int Hedef=0;  
int KalanMiktar=0;  
int icilenMiktar=0;  
private void btnArtiActionPerformed(java.awt.event.ActionEvent evt) {  
    int amount=cmbAmount.getSelectedIndex();  
    switch (amount) {  
        case 1:  
            Hedef=1500;  
            miktarArttırma();  
            break;  
        case 2:  
            Hedef=2400;  
            miktarArttırma();  
            break;  
        case 3:  
            Hedef=3000;  
            miktarArttırma();  
            break;  
        default:  
            JOptionPane.showMessageDialog(null,"Miktar giriniz!");  
            break;  
    }  
}
```

```
private void btnEksiActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    int amount=cmbAmount.getSelectedIndex();  
    if(icilenMiktar>0){  
        switch (amount) {  
            case 1:  
                Hedef=1500;  
                miktarAzaltma();  
                break;  
            case 2:  
                Hedef=2400;  
                miktarAzaltma();  
                break;  
            case 3:  
                Hedef=3000;  
                miktarAzaltma();  
                break;  
            default:  
                JOptionPane.showMessageDialog(null,"Miktar giriniz!");  
                break;  
        }  
    }  
}
```

```

private void miktarArttırma() {
    if (Hedef != icilenMiktar) {
        if (icilenMiktar == 0) {
            icilenMiktar += 150;
            lblIcilenMiktar.setText(String.valueOf(icilenMiktar));
            KalanMiktar = Hedef - icilenMiktar;
            lblKalanMiktar.setText(String.valueOf(KalanMiktar));
            sql.Ekle();
        }
        else if (icilenMiktar > 0) {
            icilenMiktar += 150;
            lblIcilenMiktar.setText(String.valueOf(icilenMiktar));
            KalanMiktar = Hedef - icilenMiktar;
            lblKalanMiktar.setText(String.valueOf(KalanMiktar));
            sql.Guncelle();
        }
    }
}

private void miktarAzaltma() {
    if (icilenMiktar == 150) {
        icilenMiktar -= 150;
        sql.Sil();
        lblIcilenMiktar.setText(String.valueOf(icilenMiktar));
        KalanMiktar += 150;
        lblKalanMiktar.setText(String.valueOf(KalanMiktar));
    }
    else {
        icilenMiktar -= 150;
        lblIcilenMiktar.setText(String.valueOf(icilenMiktar));
        KalanMiktar += 150;
        lblKalanMiktar.setText(String.valueOf(KalanMiktar));
        sql.Guncelle();
    }
}


```


Miktar arttırma ve miktar azaltmak için yazdığımız fonksiyonlar. Bu fonksiyonları ekleme ve çıkarma butonuna ekledik.

```
private void btnGetirActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    try {  
        Class.forName("com.mysql.jdbc.Driver");  
    }  
    catch(ClassNotFoundException e){  
        JOptionPane.showMessageDialog(null,"Hata:" + e.getMessage());  
    }  
    Connection baglanti = null;  
    try{  
        baglanti=DriverManager.getConnection("jdbc:mysql://localhost:3306/waterreminder","root","1999");  
    } catch (SQLException e) {  
        JOptionPane.showMessageDialog(null,"Hata:" + e.getMessage());  
    }  
  
    int Ay=cmbYil.getSelectedIndex();  
    switch (Ay) {  
        case 1:  
            try {  
                PreparedStatement uygula=baglanti.prepareStatement("SELECT Tarih, Hedef AS 'Hedefiniz',IcilenMiktari AS 'İçtiğiniz Miktar' FROM wa  
                ResultSet rs=uygula.executeQuery();  
                tblKayit.setModel(DBUtils.resultSetToTableModel(rs));  
            } catch (SQLException ex) {  
                JOptionPane.showMessageDialog(null,ex.getMessage());  
            }  
            break;
```

Verileri, comboboxta seçilen yıla göre tabloya getirmek için veritabanından çektik ve gerekli sql sorgusunu yazdık.

3.Tasarım





Hoşgeldiniz!

Günlük alınması gereken miktarın altında su tüketildiği zaman,bazı sağlık sorunlarına davetiye çıkabilir. Vücudumuzun neredeyse dörtte üçünü oluşturan su, insan sağlığının devamı için oldukça faydalı.


Günde 1,5-3 litre arası su tüketmelisiniz.

Kayıtlarınıza buradan ulaşabilirsiniz

Kayıtlar

Miktarı Seçin

Seç



-

+

İçilen Miktar:

0

Kalan Miktar:


0


Message



Su içmeyi unutmayınız!

OK





Kayıtlarınız

Miktarlar mililitre cinsinden gösterilmektedir.

Su miktarı yeterli gelmiyorsa hedef miktarını arttırabilirsiniz.

İçtiğiniz miktarın hedefe yakın olmasına mümkünse eğit olmasına özen gösterin.

Geri

2019

Veriyi Getir

Tarih	Hedefiniz	İçtiğiniz Miktar
2019-12-27	2400	1350
2019-12-25	1500	450
2019-12-22	1500	300
2019-12-21	1500	300
2019-12-18	2400	450