

Random Access File



```
class Dosya {
    RandomAccessFile personnelFile;
    DataInputStream dataIn;

    int outputBufLength = 200;
    byte outputBuf[] = new byte[outputBufLength];
    String dosyaAdi;
    int lengthOfRow;
    int k;
    Dosya(String dAdi, int satBoyu) {
        dosyaAdi = dAdi;
        lengthOfRow = satBoyu;
    }

    public boolean dosyaAc() {
        try {
            personnelFile = new RandomAccessFile(dosyaAdi,"rw");
            dataIn = new DataInputStream(System.in);
            return true;
        } catch (IOException e) { return false; }
    }
}
```

```
public void satirYaz(int row, String satir) {
    try {
        satir.getBytes(0,satir.length(),outputBuf,0);
        personnelFile.seek(row*lengthOfRow);
        personnelFile.write(outputBuf,0,satir.length());
    } catch (IOException e) { System.out.println("Dosyaya Yazamadım"); }
}
```

```
public String satirOku(int row) {
    String ret;
    int myChar;
    try {
        personnelFile.seek(row*lengthOfRow);
        personnelFile.read(outputBuf,0,lengthOfRow);
        ret = new String(outputBuf,0);
        return ret;
    } catch (IOException e) { System.out.println("Dosyayi Okuyamadım"); }
    return "";
}
```



```
public int findField(String findValue,int basPos, int uzn) { //Nesne koduna göre arama yapar
    String retStr,nKoduTmp;
    findValue = checkMaxLength(findValue,20);
    try {
        for (int i=0;i<=(personnelFile.length()/lengthOfRow);i++) {
            retStr = satirOku(i);
            nKoduTmp = retStr.substring(basPos,basPos+uzn);
            if (nKoduTmp.equals(findValue)) return i;
        }
    } catch (IOException e) {
        System.out.println("Dosya acilamadan arama islemi yapildi...");
        return -2;
    }
    return -1;
}
```





```
public int findField(String findValue,int basPos, int uzn, int offSet) { //Nesne koduna göre arama yapar
    String retStr,nKoduTmp;
    findValue = checkMaxLength(findValue,20);
    try {
        for (int i=offSet;i<=(personnelFile.length()/lengthOfRow);i++) {
            retStr = satirOku(i);
            nKoduTmp = retStr.substring(basPos,basPos+uzn);
            if (nKoduTmp.equals(findValue)) return i;
        }
    } catch (IOException e) {
        System.out.println("Dosya acilamadan arama islemi yapildi...");
        return -2;
    }
    return -1;
}
```

```
public int getLastRow() {
    try {
        return (int)(personnelFile.length()/lengthOfRow);
    } catch (IOException e) { return -2;}
}
```



```
String checkMaxLength(String s,int maxLength) {
    if (s.length() > maxLength) return s.substring(0,maxLength);

    for(int i=s.length()-1;i<=maxLength;i++) {
        if (s.length() < maxLength) s += ' ';
        if (s.length() == maxLength) return s;
    }
    return s;
}
}
```

File Viewer

```
class fileViewer {  
    FileDialog f;  
    String dizin,dosyaAdi;  
  
    fileViewer(Frame frame) {  
        f=new FileDialog(frame,"Open File",FileDialog.LOAD);  
        f.show();  
        dizin = f.getDirectory();  
        dosyaAdi=f.getFile();  
    }  
}
```



```
fdosya = new Dosya("veri.dat",140);  
    if (!fdosya.dosyaAc()) {  
        System.out.println("Dosya acilmadi...");  
    }  
}
```



To close a file write in simple form:

```
in.close();
```

```

public boolean handleEvent(Event e) {
if (e.target.equals(fmp.txtSadi)) {
    fmp.clear();
    int ret = fdosya.findField(fmp.txtSadi.getText(),0,20);
    if (ret >=0 ) {
        String s = fdosya.satirOku(ret);
        sinifadi = s.substring(0,20);
        soz1 = s.substring(1*20,1*20+20);
        soz2 = s.substring(2*20,2*20+20);
        soz3 = s.substring(3*20,3*20+20);
        soz4 = s.substring(4*20,4*20+20);
        soz5 = s.substring(5*20,5*20+20);
        soz6 = s.substring(6*20,6*20+20);

        fmp.txtSadi.setText(sinifadi);
        fmp.txtSoz1.setText(soz1);
        fmp.txtSoz2.setText(soz2);
        fmp.txtSoz3.setText(soz3);
        fmp.txtSoz4.setText(soz4);
        fmp.txtSoz5.setText(soz5);
        fmp.txtSoz6.setText(soz6);
    }
    if (ret != -1) row=ret; else row=fdosya.getLastRow();
    return true;
} else return super.handleEvent(e);
}

```




```

public boolean action(Event evt, Object obj) {
    String label = (String)obj;

    if (label.equals("Nesne Kayit")) {
        sinifadi=fmp.txtSadi.getText();
        if (sinifadi.length()==0) System.out.println("Bos deger var degeri ver");
        else {
            sinifadi=checkMaxLength(sinifadi,20);
            soz1=checkMaxLength( fmp.txtSoz1.getText(), 20 );
            soz2=checkMaxLength( fmp.txtSoz2.getText(), 20 );
            soz3=checkMaxLength( fmp.txtSoz3.getText(), 20 );
            soz4=checkMaxLength( fmp.txtSoz4.getText(), 20 );
            soz5=checkMaxLength( fmp.txtSoz5.getText(), 20 );
            soz6=checkMaxLength( fmp.txtSoz6.getText(), 20 );
            fdosya.satirYaz(row,sinifadi+soz1+soz2+soz3+soz4+soz5+soz6);
            fclearTexts();
        }
    }

    if (label.equals("Cikis")) hide();
    return true;
}

```



```
void fclearTexts() {  
    fmp.txtSadi.setText("");  
    fmp.txtSoz1.setText("");  
    fmp.txtSoz2.setText("");  
    fmp.txtSoz3.setText("");  
    fmp.txtSoz4.setText("");  
    fmp.txtSoz5.setText("");  
    fmp.txtSoz6.setText("");  
}
```

```
String checkMaxLength(String s,int maxLength) {  
    if (s.length() > maxLength) return s.substring(0,maxLength);  
    for(int i=s.length();i<=maxLength;i++) {  
        if (s.length() < maxLength) s += ' ';  
        if (s.length() == maxLength) return s;  
    }  
    return s;  
}
```

