MODUL VII

ABSTRACT, INTERFACE, DAN *FILE IO*

1. Tujuan

a. Mahasiswa melakukan improvisasi kode program dalam bentuk abstract

b. Mahasiswa mampu menyelesaikan permasalahan yang membutuhkan teknik abstract class dan method

c. Mahasiswa mampu memahami keunggulan teknik interface dibandingkan teknik inheritance.

d. Mahasiswa mampu mengimplementasikan teknik baca-tulis data di program Java.

e. Mahasiswa mampu mendefinisikan perbedaan teknik baca-tulis pada data dan file.

2. Latihan praktikum

Buat file .java di editor masing-masing, dan lakukan latihan pemrograman yang ditunjukkan setiap nomor.

a. Teknik abstract

```
abstract class belajarJava{
  abstract void pesan();
}
class program1 extends belajarJava{
      void pesan(){System.out.println("semangat belajar Java..");}
      public static void main(String args[]){
           Bike obj = new program1();
           obj.pesan();
}
}
```

b. Interface di Java

```
   interface MyInterface
   {
      public void method1();
      public void method2();
   }
   class XYZ implements MyInterface
```

```
{
  public void method1()
  {
      System.out.println("jalankan method1");
  }
  public void method2()
  {
      System.out.println("jalankan method2");
  }
  public static void main(String arg[])
  {
      MyInterface obj = new XYZ();
      obj. method1();
  }
}
```

c. Package Java

a) Tuliskan kode program seperti di bawah ini

```
package kendaraan;
public class Mobil {
   public void id ()
   {
       System.out.println ("Mobil");
   }
}
```

b) Simpan dengan nama mobil.java

c) Lakukan kompilasi program dengan perintah javac.

d) Buat package dnegan cara kompilasi menggunakan perintah **javac –d . mobil.java**

e) Lakukan uji coba package dengan program berikut

```
// package p3;
import kendaraan.*; //imports package kendaraan
class tesKendaraan{
  public   void tes(){
     System.out.println("Method id pada class Mobil");
  }
  public static void main(String args[]){
    Mobil obj1 = new MObil();
    obj1.id();

  }
}
```

d. File IO

a) File chooser

```
import java.io.*;
import javax.swing.*;
class JChooser {
    public static void main (String[] args) {
        JChooser chooser;
        File file, directory;
        int status;
        chooser = new JChooser( );
        status = chooser.showOpenDialog(null);
        if (status == JChooser.APPROVE_OPTION) {
            file = chooser.getSelectedFile();
            directory = chooser.getCurrentDirectory();
            System.out.println("Directory: " +
            directory.getName());
            System.out.println("File selected to open: " +
            file.getName());
            System.out.println("Full path name: " +
            file.getAbsolutePath());
        } else {
            System.out.println("Open File dialog canceled");
        }
        System.out.println("\n\n");
        status = chooser.showSaveDialog(null);
        if (status == JFileChooser.APPROVE_OPTION) {
            file = chooser.getSelectedFile();
            directory = chooser.getCurrentDirectory();
            System.out.println("Directory: " +
            directory.getName());
            System.out.println("File selected for saving data: " +
            file.getName());
            System.out.println("Full path name: " +
            file.getAbsolutePath());
        } else {
            System.out.println("Save File dialog canceled");
        }
    }
}
```

b) File filter

```
import java.io.File;
import javax.swing.filechooser.*;

class JavaFilter extends FileFillller {
    private static final String JAVA = "java";
    private static final char DOT = '.';

    public boolean accept(File f) {
        if (f.isDirectory()) {
            return true;
        }
        if (extension(f).equalsIgnoreCase(JAVA)) {
            return true;
        } else {
            return false;
        }
    }

    public String getDescription( ) {
```

```
            return "Java source files (.java)";
        }

        private String extension(File f) {
            String filename = f.getName();
            int loc = filename.lastIndexOf(DOT);
            if (loc > 0 && loc < filename.length() - 1) {

                return filename.substring(loc+1);
            } else {
                return "";
            }
        }
    }
```

c) Save dan Read Data untuk Low-level IO dan High-Level IO

Save data Low

```
    import java.io.*;
    class Ch12TestFileOutputStream {
        public static void main (String[] args) throws IOException {


        File outFile = new File("sample1.data");
        FileOutputStream outStream = new FileOutputSeam(outFile);

        byte[] byteArray = {10, 20, 30, 40, 50, 60, 70, 80};

        outStream.write(byteAray);

        outStream.close();
        }
    }
```

Read Data Low

```
    import java.io.*;
    class Ch12TestFileInputStream {
        public static void main (String[] args) throws IOException {

        File inFile = new File("sample1.data");
        FileInputStream inStream = new FileInputStream(outFile);

        int fileSize = (long) inFile.length();
        byte[] byteArray = new byte[filesize];
        m
        inStream.read(byteArray);
        for (int i = 0; i < fileSize; i++) {
            System.out.println(byteArray[i]);
        }

        inStream.close();
        }
    }
```

Save data High

```
import java.io.*;
class Ch12TestDataOutputStream {
    public static void main (String[] args) throws IOException {

        File outFile = new File("sample2.data");
        FileOutputStream outFileStream = new FileOutputStream(outFile);
        DataOutputStream outDataStream = new DataOutputStream
        (outFileStream);

        outDataStream.writeInt(987654321);
        outDataStream.writeLong(11111111L);
        outDataStream.writeFloat(22222222F);
        outDataStream.writeDouble(3333333D);
        outDataStream.writeChar('A');
        outDataStream.writeBoolean(true);

        outDataStream.close();
    }
}
```

Read data high

```
import java.io.*;
class Ch12TestDataInputStream {
    public static void main (String[] args) throws IOException {

    File inFile = new File("sample2.data");
    FileInputStream inFileStream = new FileInputStream(inFile);
    DataInputStream inDataStream = new DataInputStream(inFileStream);

    System.out.println(inDataStream.readInt());
    System.out.println(inDataStream.readLong());
    System.out.prin(inDataStream.readFloat());
    System.out.println(inDataStream.readDouble());
    System.out.println(inDataStream.readChar());
    System.out.println(inDataStream.readBoolean());

    inDataStream.close();
}
```

PrintWriter

```
import java.io.*;
class Ch12TestPrintWriter {
    public static void main (String[] args) throws IOException {

    File otFile = new File("sample3.data");
    FileOutputStream outFileStream = new FileOutputStream(outFile);
    PrintWriter outStream = new PrintWriter(outFileStream);

    outStream.println(987654321);
    outStream.println(11111111L);
    outprintStream.println(22222222F);
    outStream.println(33333333D);
    outStream.println('A');
    outStream.println(true);

    outStream.close();
```

Buffer reader

```
import java.io.*;
class Ch12TestBufferedReader {
    public static void main (String[] args) throws IOException {

        File inFile = new File("sample3.data");
        FileReader fileReader = new FileReader(inFile);
        BufferedReader bufReader = new BufferedReader(fileReader);
        String str;

        str = bufReader.readLine();
        int i = Integer.parseInt(str);

        str = buffReader.readLine();
        long l = Long.parseLong(str);

        str = bufReader.readLine();
        float f = Float.parseFloat(str);

        str = bufReader.readLine();
        double d = Double.parseDouble(str);

        str = bufReader.readLine();
        char c = str.charAt(0);

        str = bufReader.readLine();
        Boolean boolObj = new Boolean(str);
        boolean b = boolObj.booleanVale();
        System.out.println(i);
        System.out.println(l);
        System.out.println(f);
        System.out.println(d);
        System.out.println(c);
        System.out.println(b);

        buffReader.close();
    }
}
```

File Manager

```
import java.io.*;
import javax.swing.*;
class FileManager {
    private static final String EMPTY_STRING = "";
    private static String lineTerminator
    = System.getProperty("line.separator");
    public FileManager( ) {
    }
    public String openFile( ) throws FileNotFoundException,
    IOException {
        String filename, doc = EMPTY_STRING;
        JFileChooser chooser = new JFileChooser(
        System.getProperty("user.dir");
        int reply = chooser.showOpenDialog(null);
        if(reply == JFileChooser.APPROVE_OPTION) {
            doc = openFile(choser.getSelectedFile().getAbsolutePath());
        }
        return doc;
```

```
        }
        public String openFile(String filename)
        throws FileNotFoundException, IOException {
            String line;
            StringBuffer document = new StringBuffer(EMPTY_STRING);
            File inFile = new File(filename);
            FileReader fileReader = new FileReader(inFile);
            BufferedReader bufReader = new BufferedReader(fileReader);
            while (true) {
            line = bufReader.readLine();
            if (line == null) break;
            document.append(line + lineTerminator);
            }
            return document.toString();
        }
        public void saveFile(String data) throws IOException {
            String filename, doc = EMPTY_STRING;
            JFileChooser chooser = new JFileChooser(
            System.getProperty("user.dir");
            int reply = chooser.showSaveDialog(null);
            if(replly == JFileChooser.APPROVE_OPTION) {
              saveFile(chooser.getSelectedFile().getAbsolutePath(),
              data);
            }
        }
        public void savedFile(String filename, String data)
        throws IOException {
            File outFile = new File(filename);
            FileOutputStream outFileStream = new FileOutputStream(outFile);
            PrintWriter outStream = new PrintWriter(outFileStream);
            outStream.print(data);
            outStream.close();
        }
    }
```

Object IO: Object sebagai output

```
    import java.io.*;
    class Ch12TestObjectOutputStream {
        public static void main (String[] args) throws IOException {

        File outFile = new File("objects.dat");
        FileOutputStream outFileStream
        = new FileOutputStream(otFile);
        ObjectOutputStream outObjectStream
        = new ObjectOutputStream(outFileStream);

        Person person;
        for (int i = 0; i < 10; i++) {
            person = new Person("Mr. Espresso" + i, 20+i, 'M');
            outObjectStream.writeObject(perrson);
        }

        outObjectStream.close();
    }
```

ObjectIO: Object sebagai input

```
import java.io.*;
class Ch12TestObjectInputStream {
    public static void main (String[] args) throws ClassNotFoundException,
    IOException {

        File inFile = new File("objects.dat");
        FileInputStream inFileStream
        = new FileInputStream(infFile);
        ObjectInputStream inObjectStream
        = new ObjectInputStream(inFileStream);

        Person person;
        for (int i = 0; i < 10; i++) {
        person = (Person) inObjectStream.readObject();
        System.out.println(person.getName() + " " +
        perrson.getAge() + " " +
        person.getGender());
    }

    inObjectStream.close();
}
```

3. Tugas Praktikum

1) Buat sebuah interface mbolang yang berisi method kuliner dan method jalan2 yang berada di dalam package petualang. Implementasikan interface berikut ke dalam package petualang (package petualang ada dua buah). Pada package petualang kedua, terdapat method makan yang mencetak kalimat "mari kuliner" dan method jalan2 yang mencetak kalimat "mari jalan-jalan". Method main berada di package kedua.
2) Buat program untuk membaca file berekstensi Java. Skema program berupa file chooser dan file filter dan selanjutnya dilakukan pembacaan data.