/\* Denne klassen skal hente input fra brukeren i tillegg til å kalle opp og starte andre klasser. Den viser brukeren en valgmeny der brukeren velger hva den skal gjøre, deretter gjør den det brukeren vil. Det inneholder en del tester for å sikre at brukeren skriver inn riktig i de ulike feltene slik at programmet ikke bare kræsjer, men enten gir brukeren en ny sjanse eller lukker programmet.\*/

// importerer ulike klasser

import java.util.FormatterClosedException;

import java.util.NoSuchElementException;

import javax.swing.JOptionPane;

public class Main

{

//Gjør gruppeobjektet (med alle studentene) tilgjengelig for hele Main-metoden.

private Gruppe gruppe = null;

// sender inn grp fra klassen Gruppe

public Main(Gruppe grp)

{

//Setter den lokale gruppevariablen til gruppa som er lastet fra fila.

gruppe = grp;

//Viser menyen (starter programmet)

visMeny();

}

public void visMeny()

{

// gir brukeren ulike valg som kan taes

String valg =JOptionPane.showInputDialog(

null,

"Skriv inn tallet på hva du vil gjøre:\n"

+ "1: Legg til student.\n"

+ "2: Generer gruppe.\n"

+ "3: Endre student eller legg til oppgaver.\n"

+ "0: Avslutt.");

// sjekker om brukeren trykker på cancel og gir mulighet til å starte på nytt

if (valg == null)

{

System.exit(0);

}

if (valg.equals("1"))// hvis brukeren skriver 1 slår denne til

{

// starter opp metoden aapneFil og leggTilStudent

leggTilStudent();

}

else if (valg.equals("2"))// hvis brukeren skriver 2 slår dette til

{

// starter metoden lagDiverseGrupper-GenererGruppe fra Ida

lagDiverseGrupper();

}

else if (valg.equals("3"))

{

// kaller opp vindus klassen GUIMain til Øystein

new GUIMain(gruppe, this);

}

else if (valg.equals("0"))// hvis brukeren skriver 0 så slår dette til

{

System.exit(0); // Avslutt

}

else// sjekker om brukeren skriver inn noe feil eller trykker ok/kryss og gir brukeren mulighet til å starte på nytt

{

int ja = JOptionPane.showOptionDialog(null, "Du skrev ikke riktig, prøve igjen?",

"Feil",

JOptionPane.YES\_NO\_OPTION,

JOptionPane.QUESTION\_MESSAGE,

null,

null,

null);

if (ja == JOptionPane.YES\_OPTION)

{

// start på nytt om bruker trykker ja

visMeny();

}

else

{

// avslutt om bruker trykker nei eller cancel/kryss

System.exit(0);

}

}// slutt på else/if tester

}// slutt på visMeny-metode

public void leggTilStudent()

{

// oppretter en konstruktør til klassen Student

Student student = null;

// gjør mann tilgjengelig gjennom hele metoden

boolean mann = false;

// gjør svar tilgjengelig gjennom hele metoden

int svar;

// starter en løkke der brukeren kan skrive inn input

do

{

// kaller opp Studentklassen

student = new Student();

// fornavn fra brukeren

String elevFornavn = JOptionPane.showInputDialog(null, "Skriv inn fornavnet på eleven:", "Fornavn", JOptionPane.QUESTION\_MESSAGE);

// sjekker om brukeren trykker cancel eller kryss og avslutter programmet

if (elevFornavn == null || elevFornavn == "")

{

System.exit(0);

}

// hvis brukeren lar input stå tomt får sjans på nytt

if (elevFornavn.isEmpty())

{

do

{

elevFornavn = JOptionPane.showInputDialog(null, "Må ha et fornavn, skriv inn på nytt:");

// avslutt om det blir trykket cancel/kryss

if (elevFornavn == null || elevFornavn == "")

{

System.exit(0);

}

}while (elevFornavn.isEmpty());// fortsett så lenge input er tom

}

// input fra brukeren

String elevEtternavn = JOptionPane.showInputDialog(null, "Skriv inn etternavn på eleven:", "Etternavn", JOptionPane.QUESTION\_MESSAGE);

// sjekker om brukeren trykker cancel eller kryss avslutt program

if (elevEtternavn == null || elevEtternavn == "")

{

System.exit(0);

}

// hvis brukeren lar input stå tomt får sjans på nytt

if (elevEtternavn.isEmpty())

{

do

{

elevEtternavn = JOptionPane.showInputDialog(null, "Må ha et etternav, skriv inn på nytt:");

// sjekker om bruker trykker cancel/kryss og avslutter

if (elevEtternavn == null || elevEtternavn == "")

{

System.exit(0);

}

}while (elevFornavn.isEmpty());// fortsett så lenge input er tom

}

// input fra brukeren

String elevKjonn = JOptionPane.showInputDialog(null, "Skriv inn hvilket kjønn, mann eller dame?", "Kjønn", JOptionPane.QUESTION\_MESSAGE);

// sjekker om bruker trykker cancel/kryss og avslutter

if (elevKjonn == null || elevKjonn == "")

{

System.exit(0);

}

// hvis brukeren lar input stå tom, gi en ny sjans

if (elevKjonn.isEmpty())

{

do

{

elevKjonn = JOptionPane.showInputDialog(null, "Må ha et kjønn, skriv inn på nytt:");

// hvis bruker trykker cancel/kryss avslutt

if (elevKjonn == null || elevKjonn == "")

{

System.exit(0);

}

}while (elevKjonn.isEmpty());// fortsett så lenge input er tom

}

// sjekker om brukeren har skrevet inn mann

if (elevKjonn.toLowerCase().equals("mann"))

mann = true;// setter da mann til true

// sjekker om brukeren har skrevet inn noe annet enn mann

else

mann = false;// setter da mann til false

// input fra brukeren

String studieStart = JOptionPane.showInputDialog(null, "Hvilket år startet studenten?", "Studiestart", JOptionPane.QUESTION\_MESSAGE);

// hvis brukeren trykker cancel skriv ut beskjed og ny input

if (studieStart == null || studieStart == "")

{

System.exit(0);

}

// hvis brukeren lar input stå tom start på nytt

if (studieStart.isEmpty())

{

do

{

studieStart = JOptionPane.showInputDialog(null, "Må ha et studieår, skriv inn på nytt:");

if (studieStart == null || studieStart == "")

{

System.exit(0);

}

}while (studieStart.isEmpty());// fortsett så lenge input er tom

}

int studStart = Integer.parseInt(studieStart);// parse til integer

// hvis input er mindre enn 2010 eller større enn 2014

if (studStart < 2010 || studStart > 2014)

{

do

{

studieStart = JOptionPane.showInputDialog(null, "Studenter som går på dette studiet har startet mellom 2010 og 2014, skriv inn årstall på nytt.");

// hvis bruker trykker cancel/kryss avslutt

if (studieStart == null || studieStart == "")

{

System.exit(0);

}

studStart = Integer.parseInt(studieStart);

}while (studStart < 2010 || studStart > 2014);// fortsett så lenge input er mindre enn 2010 eller større enn 2014

}

// input fra brukeren

String fagomrade = JOptionPane.showInputDialog("Velg nr. for faget til studenten:\n"

+ "1. Norsk\n"

+ "2. Engelsk\n"

+ "3. Matematikk");

// hvis bruker trykker cancel/kryss avslutt

if (fagomrade == null || fagomrade == "")

{

System.exit(0);

}

// hvis input står tom, gi en ny sjans

if (fagomrade.isEmpty())

{

do

{

fagomrade = JOptionPane.showInputDialog("Må ha et fag, prøv på nytt.\n"

+ "1. Norsk\n"

+ "2. Engelsk\n"

+ "3. Matematikk");

if (fagomrade == null || fagomrade =="")// hvis kryss/cancel avslutt

{

System.exit(0);

}

}while (fagomrade.isEmpty());// fortsett så lenge input er tom

}

int fag = Integer.parseInt(fagomrade);// parse til integer

// hvis bruker har tatt et tall utenfor intervallet, start på nytt

if (fag < 1 || fag > 3)

{

do

{

JOptionPane.showMessageDialog(null, "Feil nr, prøv igjen.");

fagomrade = JOptionPane.showInputDialog("Velg nr. for faget til studenten:\n"

+ "1. Norsk\n"

+ "2. Engelsk\n"

+ "3. Matematikk");

if (fagomrade == null || fagomrade == "")// hvis kryss/cancel, avslutt

{

System.exit(0);

}

fag = Integer.parseInt(fagomrade);

}while (fag < 1 || fag > 3);// fortsett så lenge tallet er utenfor intervallet

}

// slutt på input fra brukeren

// start å sende input fra brukeren til klassen Student for senere arbeid

try

{

student.setFornavn(elevFornavn);//

student.setEtternavn(elevEtternavn);

student.setKjonn(mann);

student.setStudiestart(studieStart);

if (fagomrade.equals("1"))

{

student.setFag("Norsk");// her bestemmes det hvilket fag studenten har

}

else if (fagomrade.equals("2"))

{

student.setFag("Engelsk");// her bestemmes det hvilket fag studenten har

}

else if (fagomrade.equals("3"))

{

student.setFag("Matematikk");// her bestemmes det hvilket fag studenten har

}

// sender studentene til klassen Gruppe for å gjøre til objekter

gruppe.leggTilStudent(student);

// etter hver gang leggTilStudent har kjørt så sendes de inn i filbehandlings-klassen og lagres i objekter der

Filbehandling.lagreGruppe(gruppe);

}

catch (FormatterClosedException fce)// hvis try ikke slår til - feilmelding

{

JOptionPane.showMessageDialog(

null,

"",

"Feil ved skriving til fil",

JOptionPane.PLAIN\_MESSAGE);

return;

}

catch (NoSuchElementException ee)// hvis try ikke slår til - feilmelding

{

JOptionPane.showMessageDialog(

null,

"",

"Feil input, prøv igjen",

JOptionPane.PLAIN\_MESSAGE);// feilmelding

}

// brukeren får valg om å starte innlesing av flere elever på nytt

svar = JOptionPane.showOptionDialog(null,

"Vil du lese inn flere elever?",

"Klasseliste",

JOptionPane.YES\_NO\_OPTION,

JOptionPane.QUESTION\_MESSAGE,

null,

null,

null);

if (svar == JOptionPane.NO\_OPTION)

{

visMeny();

}

}while (svar == JOptionPane.YES\_OPTION);

System.exit(0);

}// slutt på metoden leggTilStudent

public void lagDiverseGrupper()

{

//lokal variabel

int igjen;

// start en løkke for å finne ut hvilke grupper brukeren vil generere

do

{

// hvis et valg-tre som sender brukeren videre på riktig gren

String valgtre = JOptionPane.showInputDialog(null,

"Skriv inn tallet på hva du vil gjøre:\n"

+ "1: Sorter etter kjønn.\n"

+ "2: Sorter etter fag.\n"

+ "3: Sorter etter studiestart.\n"

+ "4: Sorter alfabetisk.\n"

+ "5: Lag tilfeldig sammensetting.\n"

+ "6: Lag liste over de som er godkjent til eksamen.\n"

+ "0: Avslutt.");

// kaller opp og konstruerer en konstruktør for klassen GenererGruppe

GenererGruppe kjor = new GenererGruppe(gruppe);

// sortere etter kjønn

if (valgtre.equals("1"))

{

// start metoden sortereKjonn i klassen GenererGruppe (Ida)

kjor.sortereKjonn();

}

// sortere etter fag

else if (valgtre.equals("2"))

{

// start metoden sortereFag i klassen GenererGruppe (Ida)

kjor.sortereFag();

}

// sortere etter studiestartår

else if (valgtre.equals("3"))

{

// start metoden sortereStudiestart i klassen GenererGruppe (Ida)

kjor.sortereStudiestart();

}

// sorterer alfabetisk

else if (valgtre.equals("4"))

{

// start metoden sortereAlfabetisk i klassen GenererGruppe (Ida), sender inn studenter fra klassen Gruppe

kjor.sortereAlfabetisk(gruppe.hentStudenterAsArray());

}

// sett sammen tilfeldig

else if (valgtre.equals("5"))

{

// start metoden sortereTilfeldig i klassen GenererGruppe (Ida), sender inn studenter fra klassen Gruppe

kjor.sortereTilfeldig(gruppe.hentStudenterAsArray());

}

// sorterer etter de som kan gå til eksamen

else if (valgtre.equals("6"))

{

// start metoden sortereGodkjent i klassen GenererGruppe (Ida)

kjor.sortereGodkjent();

}

// avslutter

else if (valgtre.equals("0"))

{

System.exit(0);

}

// hvis det er skrevet feil tall eller en bokstav så får bruker ny sjans

else

{

igjen = JOptionPane.showOptionDialog(null,

"Du skrev inn et valg som ikke finnes, prøve på nytt?",

"Lister",

JOptionPane.YES\_NO\_OPTION,

JOptionPane.QUESTION\_MESSAGE,

null,

null,

null);

// hvis brukeren velger ja, start gruppevalg igjen

if (igjen == JOptionPane.YES\_OPTION)

lagDiverseGrupper();

// hvis brukeren gjør noe annet start hele programmet på nytt

else

{

visMeny();

}

}

// brukeren får valg om å lage nye lister

igjen = JOptionPane.showOptionDialog(null,

"Vil du generere nye lister?",

"Lister",

JOptionPane.YES\_NO\_OPTION,

JOptionPane.QUESTION\_MESSAGE,

null,

null,

null);

// hvis brukeren trykker nei start hele programmet på nytt

if (igjen == JOptionPane.NO\_OPTION)

{

visMeny();

}

}while(igjen == JOptionPane.YES\_OPTION);// fortsett løkke med å generere grupper så lenge igjen er ja

System.exit(0);

}// slutt på metoden lagDiverseGrupper

}// slutt på klassen Main