Отговори на примерния вариант:

2.

Решение:

```
public class Age {
 static void controlAge(int age) {
 if (age < 18) {
 throw new ArithmeticException("Достъпа е забранен под 18 години");
 }
 else { System.out.println("Достъпа е позволен");
 }
 }
  public static void main(String[] args) {
    controlAge(16);
  }
}
```

3.Отговор

г) привет изключение хванато накрая иключение и след изключение

4.Как се нарича класа от JavaFX, които представя целия прозорец на приложението.

b)javafx.stage.Stage

5.

Решение:

```
import javax.swing.*;
 import java.awt.*;
 import java.awt.event.*;
 import java.util.ArrayList;

 public class InterfaceStudent extends JFrame{
    protected JButton ad;
    protected JTextField tf[]= new JTextField[3];
    protected JTextArea slist;

    protected JPanel contr,plst;


    protected AdSt adSt;
  private ArrayList<Student> prs=new ArrayList<Student>(10);

    InterfaceStudent(int x, int y, int ln, int ht){

      this.setLayout(new BorderLayout());
      this.setBounds(x, y, ln, ht);
      this.setVisible(true);
      this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
      this.setTitle("Студенти");
      slist = new JTextArea(5, 10);
      slist.setEditable(false);

      contr = new JPanel(new FlowLayout());
      plst = new JPanel(new FlowLayout());

      ad = new JButton("добавете");
      contr.add(ad);
      tf[0]= new JTextField("име",10);
      tf[1]= new JTextField("курс",9);
      tf[2]= new JTextField("оценка",4);
      contr.add(tf[0]);
      contr.add(tf[1]);
       contr.add(tf[2]);
      ad.addActionListener(adSt=new AdSt());
      tf[1].addActionListener(adSt);
      plst.add(new JScrollPane(slist));
      add("North",contr);
      add("Center",plst);
      revalidate();
          }
    class AdSt implements ActionListener{
      public void actionPerformed(ActionEvent e ){
          Student s;
          int nt;
          String n=tf[0].getText();
          String nm=tf[1].getText();
          try{
              nt=Integer.parseInt(tf[2].getText());
          }
          catch(NumberFormatException ex){
              tf[2].setText("оценка");
              return;
          }
          prs.add(s=new Student(n,nm,nt));
          tf[0].setText("име");tf[1].setText("курс");tf[2].setText("оценка");
```

```
            slist.append(s+"\n");

            slist.setCaretPosition(slist.getDocument().getLength());


            revalidate();

        }

    }

    public static void main(String [] arg){

        new InterfaceStudent(20,20,400,300);

    }

 }

class Student {


    String name; String coursename;int grade;

    Student(String name,String coursename, int grade){

        this.name = name;

        this.coursename=coursename;

        this.grade = grade;}

    public String toString(){

        return name+"   "+coursename+" "+grade;

    }

 }
```

Задача: Решение

```
 import javax.swing.*;

 import javax.swing.border.*;

 import java.awt.*;

 import java.awt.event.*;

 import java.util.*;
```

```
public class Person  extends JFrame implements ActionListener{
private JPanel  p1,p2;
private JLabel l1,l2,l3;
private JTextField field1, field2,field3;
private JTextArea area;
private JScrollPane scroll;
private JButton add,clear;
private ArrayList<Student> students;
private JButton list;
private JButton max;
public Person(){
super("Студенти");
this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE) ;
Container container = this.getContentPane();
p1= new JPanel();
 p1.setLayout(new GridLayout(2,2));
p1.setBorder(new TitledBorder(new EtchedBorder(),"Студенти"));
l1= new JLabel("Име");
p1.add(l1);
field1= new JTextField(10);
field1.addActionListener(this);
p1.add(field1);
l2=new JLabel("Успех");
p1.add(l2);
field2= new JTextField(10);
p1.add(field2);
}

container.add(p1,BorderLayout.PAGE_START);

area= new JTextArea(5,10);

scroll= new JScrollPane(area, JScrollPane.VERTICAL_SCROLLBAR_ALWAYS,
JScrollPane.HORIZONTAL_SCROLLBAR_ALWAYS);

container.add(scroll,BorderLayout.CENTER);

p2=new JPanel();

add= new JButton("Добави");

add.addActionListener(this);

p2.add(add);

clear= new JButton("Изчисти");

clear.addActionListener(this);

p2.add(clear);

max= new JButton("Максимален успех");

max.addActionListener(this);

p2.add(max);

container.add(p2,BorderLayout.PAGE_END);

students= new ArrayList<Student>();

this.setSize(450,250);

this.setVisible(true);


list=new JButton("Списък");

list.addActionListener(this);

p2.add(list);

container.add(p2,BorderLayout.PAGE_END);




}

public void actionPerformed(ActionEvent e){
```

```
Object source= e.getSource();

if(source==add){

try{

Student newStudent= new Student(field1.getText(),Double.parseDouble(field2.getText()));

students.add(newStudent);

area.setText("");

area.append("Добавен студент\n"+ newStudent.toString()+"\n");

}

catch(NumberFormatException ex){

area.setText("");

area.append("Грешка при въвеждане на данните\n");

}

}

else if(source== clear){

field1.setText("");

field2.setText("");

area.setText("");

}

else if(source==field1){

String searchName=field1.getText();

boolean flag= false;

area.setText("");

area.append("Данни за "+ searchName+"\n");

for(Student c: students)

if(c.getName().equals(searchName)){

flag=true;

area.append(c.toString()+"\n");
}
if(!flag){
area.setText("");
area.append("Липсват данни за "+searchName+"\n");
}
}
else if(source==max){
   area.setText("");
   Student n=null;
   boolean flag=false;
   for(Student c: students){
      if(!flag){
         n=c;
         flag=true;
      }
      else if(c.getMark()>n.getMark())
   n=c;
         }
area.append("Студент с максимален успех среден успех\n"+n.toString()+"\n");
}
else if(source==list){
   area.setText("");
   area.append("Списък на студенти\n");
   for(Student c:students)
      area.append(c.toString()+"\n");
}
}
```

```
public static void main(String[] args){

SwingUtilities.invokeLater(new Runnable(){

public void run(){

new Person();

}

        });

}


}


public class Student {


private String name;

private double mark;

public Student(String n, double m){

name=n;

mark=m;

}

public double getMark(){

return mark;

}

public String getName(){

return name;

}


public String toString(){
```

```
return name+ "\t"+ mark;

}

}
```