

7.

Tehtävä 1.

```
public class Rannekello {
    private int tunnit;
    private int minuutit;
    public Rannekello() {
        tunnit = 0;
        minuutit = 0;
    }
    public Rannekello(int uudetTunnit, int uudetMinuutit) {
        boolean ok = asetaAika(uudetTunnit, uudetMinuutit);
        if (ok) {
            this.tunnit = uudetTunnit;
            this.minuutit = uudetMinuutit;
        }
    }
    public boolean asetaAika(int tunnit, int minuutit) {
        if (tunnit >= 0 && tunnit <= 23 && minuutit >= 0 && minuutit <= 59) {
            this.tunnit = tunnit;
            this.minuutit = minuutit;
            return true;
        } else {
            return false;
        }
    }
    public void siirraMinuutillaEteenpain() {
        if (minuutit != 59) {
            this.minuutit = this.minuutit + 1;
        } else {
            if (tunnit == 23) {
                this.tunnit = 0;
                this.minuutit = 0;
            } else {
                this.tunnit = this.tunnit + 1;
                this.minuutit = 0;
            }
        }
    }
    public int tunnit() {
        return this.tunnit;
    }
    public int minuutit() {
        return this.minuutit;
    }
}
```

Tehtävä 2.

```
public class Heratyskello {
    private int tunnit;
    private int minuutit;
    private int heratystunnit;
    private int heratysminuutit;
    public Heratyskello() {
        tunnit = 0;
        minuutit = 0;
        heratystunnit = 0;
        heratysminuutit = 0;
    }
    public Heratyskello(int uudetTunnit, int uudetMinuutit) {
        boolean ok = asetaAika(uudetTunnit, uudetMinuutit);
        if (ok) {
            this.tunnit = uudetTunnit;
            this.minuutit = uudetMinuutit;
            this.heratystunnit = 0;
            this.heratysminuutit = 0;
        }
    }
    public boolean asetaAika(int tunnit, int minuutit) {
        if (tunnit >= 0 && tunnit <= 23 && minuutit >= 0 && minuutit <= 59) {
            this.tunnit = tunnit;
            this.minuutit = minuutit;
            return true;
        } else {
            return false;
        }
    }
    public boolean asetaHeratys(int tunnit, int minuutit) {
        if (tunnit >= 0 && tunnit <= 23 && minuutit >= 0 && minuutit <= 59) {
            this.heratystunnit = tunnit;
            this.heratysminuutit = minuutit;
            return true;
        } else {
            return false;
        }
    }
    public void siirraMinuutillaEteenpain() {
        if (minuutit != 59) {
            this.minuutit = this.minuutit + 1;
        } else {
            if (tunnit == 23) {
                this.tunnit = 0;
                this.minuutit = 0;
            } else {
                this.tunnit = this.tunnit + 1;
                this.minuutit = 0;
            }
        }
    }
}
```

```

    }
}
}
public int heratystunnit() {
    return this.heratystunnit;
}
public int heratysminuutit() {
    return this.heratysminuutit;
}
public int tunnit() {
    return this.tunnit;
}
public int minuutit() {
    return this.minuutit;
}
public String toString() {
    if (this.tunnit == this.heratystunnit && this.minuutit == this.heratysminuutit) {
        return this.tunnit + ":" + this.minuutit + " herätys " + this.heratystunnit + ":" + this.heratysminuutit + " " + "PIIP!";
    } else {
        return this.tunnit + ":" + this.minuutit + " herätys " + this.heratystunnit + ":" + this.heratysminuutit;
    }
}
}

```

Tehtävä 3.

```

public class Saahavaintoasema {
    private int lampotila;
    private int suunta;
    private int nopeus;
    private boolean mitattu;
    public Saahavaintoasema() {
        this.mitattu = false;
    }
    public boolean mittaus(int lampotila, int suunta, int nopeus) {
        if (lampotila >= -50 && lampotila <= 50 && suunta >= 0 && suunta <= 359 && nopeus >= 0 && nopeus <= 100) {
            this.lampotila = lampotila;
            this.suunta = suunta;
            this.nopeus = nopeus;
            this.mitattu = true;
            return true;
        } else {
            this.mitattu = false;
            return false;
        }
    }
    public String toString() {
        if (!mitattu) {
            return "Ei mitattu";
        } else {

```

```
        return "Lämpötila " + this.lampotila + " astetta, tuulen suunta " + this.suunta + " astetta ja nopeus " + this.nopeus  
        + " m/s";  
    }  
}
```

Tehtävä 4.

```
public class Saahavaintoasema {  
    private int lampotila;  
    private int suunta;  
    private int nopeus;  
    private int maxLampotila = -51;  
    private int maxNopeus;  
    private boolean mitattu;  
    public Saahavaintoasema() {  
        this.mitattu = false;  
    }  
    public boolean mittaus(int lampotila, int suunta, int nopeus) {  
        if (lampotila >= -50 && lampotila <= 50 && suunta >= 0 && suunta <= 359 && nopeus >= 0 && nopeus <= 100) {  
            if (lampotila > this.maxLampotila) {  
                this.maxLampotila = lampotila;  
            }  
            if (nopeus > this.maxNopeus) {  
                this.maxNopeus = nopeus;  
            }  
            this.lampotila = lampotila;  
            this.suunta = suunta;  
            this.nopeus = nopeus;  
            this.mitattu = true;  
            return true;  
        } else {  
            this.mitattu = false;  
            return false;  
        }  
    }  
    public void nollaa() {  
        this.maxLampotila = this.lampotila;  
        this.maxNopeus = this.nopeus;  
    }  
    public String toString() {  
        if (!mitattu) {  
            return "Ei mitattu";  
        } else {  
            return "Lämpötila " + this.lampotila + " astetta (max. " + this.maxLampotila + "), tuulen suunta " + this.suunta + "  
            astetta ja nopeus " + this.nopeus + " m/s (max. " + this.maxNopeus + ")";  
        }  
    }  
}
```

Tehtävä 5.

```
public boolean mittarinMittaus(Saamittari mittaaaja) {  
    boolean ok = mittaus(mittaaaja.lampotila(), mittaaaja.tuulenSuunta(), mittaaaja.tuulenNopeus());  
    return ok;  
}
```

Tehtävä 6.

```
import org.junit.After;  
import org.junit.AfterClass;  
import org.junit.Before;  
import org.junit.BeforeClass;  
import org.junit.Test;  
import static org.junit.Assert.*;  
  
public class KokonaislukuTest {  
    Kokonaisluku ai;  
  
    @Before  
    public void setUp() {  
        ai = new Kokonaisluku();  
    }  
  
    @Test  
    public void lukemaAlussaNolla() {  
        assertEquals(0, ai.getLuku());  
    }  
  
    @Test  
    public void lisataanYksi() {  
        ai.setLuku(1);  
        assertEquals(1, ai.getLuku());  
    }  
  
    @Test  
    public void toStringToimii() {  
        ai.setLuku(-1);  
        assertEquals("Luku: -1", ai.toString());  
    }  
}
```

Mikäli Kokonaisluku-luokka ei hyväksyisi negatiivisia arvoja olisi testattava, ettei getLuku palauta eri arvoa kuin nolla, kun setLuvulle annetaan negatiivinen arvo.

Esim.

```
@Test  
public void negatiivistaEiHyvaksyta() {  
    ai.setLuku(-1);  
}
```

```
    assertEquals(0, ai.getLuku());  
}
```

8.

Tehtävä 1.

```
import java.util.Collections;  
import java.util.ArrayList;  
public class ArrayListString {  
    public static void main(String[] args) {  
        ArrayList<String> kokoelma = new ArrayList<>();  
        kokoelma.add("asia");  
        kokoelma.add("talo");  
        kokoelma.add("mitä");  
        kokoelma.add("haloo");  
        kokoelma.add(0, "alkuun");  
        Collections.sort(kokoelma);  
        System.out.println(kokoelma);  
        kokoelma.remove(1);  
        System.out.println(kokoelma);  
    }  
}
```

Tehtävä 2.

```
import java.util.ArrayList;  
import java.util.Collections;  
import java.util.Scanner;  
public class ArrayListInteger {  
    public static void main(String[] args) {  
        ArrayList<Integer> lista = new ArrayList<>();  
        Scanner lukija = new Scanner(System.in);  
        String stringLuku;  
        int luku;  
        char lopetus;  
        do {  
            System.out.println("Anna luku: ");  
            stringLuku = lukija.nextLine();  
            lopetus = stringLuku.charAt(0);  
            if (lopetus != 'x') {  
                luku = Integer.parseInt(stringLuku);  
                lista.add(luku);  
            }  
        } while (lopetus != 'x');  
        Collections.sort(lista);  
        System.out.println(lista);  
    }  
}
```

```

    int i;
    double summa = 0;
    for (i = 0; i < lista.size(); i++) {
        summa += lista.get(i);
    }
    int jakaja = lista.size();
    System.out.println("Keskiarvo on: " + summa / jakaja);
}
}

```

Tehtävä 3.

```

import java.util.*;
public class Laskurilista {

    public static void main(String[] args) {
        ArrayList<Laskuri> luettelo = new ArrayList<>();
        Laskuri laskuri1 = new Laskuri();
        Laskuri laskuri2 = new Laskuri();
        Laskuri laskuri3 = new Laskuri();
        luettelo.add(laskuri1);
        luettelo.add(laskuri2);
        luettelo.add(laskuri3);
        Scanner lukija = new Scanner(System.in);
        int luku;
        do {
            System.out.println("Anna luku 0, 1 tai 2. -1 lopettaa.");
            luku = Integer.parseInt(lukija.nextLine());
            if (luku >= 0 && luku <= 2) {
                System.out.println("Syötit luvun " + luku);
                if (luku == 0) {
                    laskuri1.lisaa();
                }
                if (luku == 1) {
                    laskuri2.lisaa();
                }
                if (luku == 2) {
                    laskuri3.lisaa();
                }
            } else {
                if (luku != -1) {
                    System.out.println("Ei kelpaa..");
                }
            }
        } while (luku != -1);
        System.out.println("Lukua 0 " + laskuri1.lukema() + " kpl.");
        System.out.println("Lukua 1 " + laskuri2.lukema() + " kpl.");
        System.out.println("Lukua 2 " + laskuri3.lukema() + " kpl.");
    }
}

```

Tehtävä 4.

```
import java.util.ArrayList;
import java.util.Scanner;
public class Arvosanalista {
    public static void main(String[] args) {
        ArrayList<Arvosana> arvosanalista = new ArrayList<>();
        Scanner lukija = new Scanner(System.in);
        String stringArvosana;
        int lukuArvosana;
        do {
            System.out.println("Anna arvosana: ");
            stringArvosana = lukija.nextLine();
            lukuArvosana = Integer.parseInt(stringArvosana);
            Arvosana luku1 = new Arvosana();
            boolean ok = luku1.asetta(lukuArvosana);
            if (ok) {
                luku1.asetta(lukuArvosana);
                arvosanalista.add(luku1);
            } else {
                System.out.println("Virheellinen arvosana");
            }
        } while (arvosanalista.size() <= 4);
    }
}
```