

# D Kalenteri

---

Samu Warinowski & Miika Alikirri

Kalenteri toteutetaan luomalla perinteinen kalenterinäkymä käyttäen [javax.swing](#) kirjastoa. Kalenterin tila tallennetaan Sqlite tietokantaan.

## Käyttöliittymän ratkaisuperjaate

---

Kalenterinäkymästä luodaan perinteinen näkymä jossa näytetään yhden kuukauden päivät kerrallaan. Jokasella päivällä on alue joka näyttää kuinka monta tehtävää päivälle on, ja kuinka monta niistä ollaan suoritettu. Aluetta painamalla tuodaan näkyviin näkymä jossa voi poistaa, lisätä ja muokata tehtäviä.

Jokaiselle tehtävälle asetetaan kellonaika jolloin tehtävän informaatio hälytetään käyttäjälle.

## Tietokannan ja datan käsittelyn ratkaisuperjaate

---

### Tietokanta

Kalenterissa olevien tehtävien tallentaminen Sqlite tietokantaan ratkaistaan käyttäen tietokannan mallintamista luokkien avulla. Projektiin toteutetaan abstrakti luokka "Model" jonka avulla perivätkä luokat voivat mallintaa tietokannan taulua. Luokan "Model" perii luokka "Task" joka sisältää tarvittavat rivit tehtävien mallintamiseksi.

Tietokannan rivien käsittelemiseksi Task luokka käyttää geneeristä Value luokka luokanmuuttujille. Value luokan avulla saadaan tarkistettua onko arvoa muutettu alustamisen jälkeen jolloin tietokantaan ei tarvitse päivittää kuin muutetut arvot.

Taulun "task" Sqlite malli (schema):

```
CREATE TABLE task (month INTEGER, day INTEGER, year INTEGER,
alertHour INTEGER, alertMinute INTEGER, isDone BOOLEAN, task TEXT,
id INTEGER PRIMARY KEY AUTOINCREMENT );
```

### Datan käsittely

Ohjelmaan toteutetaan TaskProvider luokka joka synkronoi datan ohjelman ja tietokannan

välillä.

## Toteutettavat luokat

Toteutuksen kannalta tärkeiden luokkien metodien määritelmät

### Data luokat

#### ***public class DatabaseHelper***

Singleton luokka sqlite tietokannan yhteydelle.

Metodin määritelmä	Kuvaus
<code>public static DatabaseHelper getInstance()</code>	Palauta Databasehelper singletonin instanssi
<code>public void execute(String cmd) throws SQLException</code>	Aja sql komento
<code>public void safeExecute(String statement, Value&lt;?&gt;[] values) throws SQLException</code>	Aja sql turvallisesti. Tämä estää sql injektioita
<code>public ResultSet query(String cmd) throws SQLException</code>	Palauta tietokannasta tietueen mukainen ResultSet

#### ***public class Value<T>***

Geneerinen luokka joka vastaa tietokanta taulun riviä.

Luokkamuuttuja	Kuvaus
<code>private T value</code>	Value luokan arvo
<code>private T originalValue</code>	Value luokan alkuperäinen arvo
<code>private String columnName</code>	Tietokannan rivin nimi

Metodin määritelmä	Kuvaus
<code>public boolean isModified()</code>	Onko luokan arvoa muutettu alustamisen jälkeen?

#### ***abstract class Model***

Abstrakti luokka tietokantataulu luokkien toteuttamiseksi.

Luokkamuuttuja	Kuvaus
private String tableName	tietokanta taulun luokan nimi
protected Value<Integer> id	"INTEGER PRIMARY KEY AUTOINCREMENT" jonka kaikki taulut toteuttavat

Metodin määritelmä	Kuvaus
protected void addIntToSchema(String columnName)	Lisää int tyyppinen rivi tauluun
protected void addBoolToSchema(String columnName)	Lisää boolean tyyppinen rivi tauluun
protected void addStringToSchema(String columnName)	Lisää String tyyppinen rivi tauluun
protected void createTable()	Luo taulu lisättyjen rivien perusteella
protected abstract void initialize()	Määrittele mallin taulun alustaminen
public void create()	Lisää luokka tietokantaan
public void update()	Päivitä muokatut arvot tietokantaan entuudestaan lisätystä luokasta
public void delete()	Poista luokka tietokannasta
protected abstract Value<?>[] getDbValues()	Palauta lista tietokannassa käytettävistä arvoista
public abstract void initFromResultSet(ResultSet rs)	Alusta luokka tietokannasta saadun RestulSet:n perusteella

### **public class Models**

Luokka tietokantojen alustamiseen kun ohjelma käynnistetään.

Luokkamuuttuja	Kuvaus
private static final Model[] MODELS	Lista alustettavista luokista

Metodin määritelmä	Kuvaus
--------------------	--------

Metodin määritelmä	Kuvaus
public static void initializeModels()	Kutsu MODELS listassa olevien luokkien initialize metodeja

***public class Task extends Model implements Comparable<Task>***

Kalenterissa olevien tehtävien tietokanta malli luokka, joka on järjestettävissä luokkamuuttujien alertHour ja alertMinute mukaan.

Luokkamuuttuja	Kuvaus
private Value<Integer> day	Task taulun day rivi.
private Value<Integer> year	Task taulun year rivi.
private Value<Integer> month	Task taulun month rivi.
private Value<Integer> alertHour	Task taulun alertHour rivi.
private Value<Integer> alertMinute	Task taulun alertMinute rivi.
private Value<String> task	Task taulun task rivi.
private Value<Boolean> isDone	Task taulun isDone rivi.

Metodin määritelmä	Kuvaus
public int compareTo(Task task)	Vertaa task luokkia keskenään alertHour ja alertMinute perusteella.

***abstract class AbstractProvider<V>***

Geneerinen abstrakti luokka objelman datan tarjoajille.

Luokkamuuttuja	Kuvaus
protected ArrayList<V> views	Lista tarjoajaa kuuntelevista luokista

Metodin määritelmä	Kuvaus
public void addView(V view)	Lisää kuuntelija
public void deleteView(V view)	Poista kuuntelija
protected abstract void notifyListeners()	Ilmoita kuuntelijoille datan muuttumisesta

***public class MonthProvider extends AbstractProvider<MonthView>***

Ohjelmassa olevan kuukauden ajan tarjoaja. Luokka on singleton tyyppinen.

Metodin määritelmä	Kuvaus
public void nextMonth()	Vaihda aika seuraavaan kuukauteen
public void previousMonth()	Vaihda aika edeltävään kuukauteen

***public class TaskProvider extends AbstractProvider<TaskView>***

Ohjelmassa olevan tehtävien tarjoaja. TaskProvider luokan tila tallennetaan automaattisesti tietokantaan. Luokka on singleton tyyppinen.

Luokkamuuttuja	Kuvaus
private HashMap<Integer, ArrayList<Task>> tasks	Kuukaudessa olevat tehtävät, avain on päivää vastaava Integer

Metodin määritelmä	Kuvaus
public void addNewTask(Task task)	Lisää uus tehtävä
public void removeTask(Task task)	Poista tehtävä
public void updateTask(int dayIndex, Task task)	Päivitä päivän listassa oleva tehtävä
public ArrayList<Task> getDayTasks(Integer day)	Hae tietyn päivän tehtävät
public Task getDayTask(Integer day, int dayIndex)	Hae tietyn päivän tietyn tietyn indeksin tehtävä
public void loadMonthData(int month, int year)	Lataa tietokannasta task muuttujaan tehtävät

**Ui luokat*****public class AlertFrame extends JFrame***

AlertFrame piirtää tehtävän kun tehtävän aika on valmis.

Konstruktori	Kuvaus
--------------	--------

Konstruktori	Kuvaus
public AlertFrame(Task task)	Argumenttina tehtävä joka halutaan näyttää

***public class MonthFrame extends JFrame***

MonthFrame piirtää ohjelman kuukausinäkymän.

***public class DayFrame extends JFrame***

DayFrame piirtää yhden päivän tehtävät. DayFrame avataan MOnthFrame kuukausinäkymästä.