

# Examen Programmeren II

Let op: de originele vragen van het examen kunnen lichtjes verschillen.

Enkele voorafgaande opmerkingen:

- Je moet verplicht drie kwartier blijven, en krijgt maximum 3 uur (wat ruim voldoende moet zijn).
- Het examen is schriftelijk (dus niet op de computer).
- Het gebruik van GSMs, rekentoestellen, ... is verboden.
- Je mag de hulpklasse *Model* uit de lessen gebruiken als dat nodig is.
- Je mag geen examenbladen of kladbladen meenemen.

Je krijgt de volgende *Enum* gegeven en zal die ook nodig hebben in de opgaven.

```
public enum Weekdag {  
  
    ZONDAG, MAANDAG, DINSDAG, WOENSDAG,  
    DONDERDAG, VRIJDAG, ZATERDAG;  
  
    @Override  
    public String toString() {  
        return name().toLowerCase();  
    }  
  
    public Weekdag next() {  
        return values()[ (ordinal()+1) % 7];  
    }  
  
    public Weekdag previous() {  
        return values()[ (ordinal()+6) % 7];  
    }  
  
}
```

1. Implementeer *WeekdagSpinnerModel* zodat een *JSpinner* doorlopend de weekdays van zondag tot zaterdag (en daaropvolgend weer zondag) kan navigeren.

Hier krijg je alvast de methoden uit de Java API voor *SpinnerModel*:

```
public interface SpinnerModel {  
  
    Object getValue() ;  
    void setValue(Object value);  
  
    Object getNextValue() ;  
    Object getPreviousValue() ;  
  
    void addChangeListener(ChangeListener l) ;  
    void removeChangeListener(ChangeListener l) ;  
  
}
```

2. Overschrijf de *toString*-methode van *Weekdag* zodanig dat deze de eerste letter in hoofdletter en de rest in kleine letters teruggeeft. Ook moet deze aangepast worden aan de taal van het besturingssysteem. Indien de taal van het besturingssysteem niet gesupport wordt, moeten de Engelse namen weergegeven worden.

A. Schrijf hier uw antwoord. Enkel de dingen die die je hebt ingevoegd of veranderd, moet je opschrijven (dus niet opnieuw de hele *Weekdag*).

B. Schrijf de naam op van de Engelse versie van het bestand.

C. Schrijf de naam op van de Nederlandse versie van het bestand.

D. Waar bevinden deze bestanden zich tijdens het uitvoeren?  
Schrijf het pad op van de map die de i18n-bestanden bevat.

3. Zorg ervoor dat wanneer de gebruiker “ctrl 0” (nul) indrukt, de lijst automatisch verspringt naar zondag. Zorg er ook voor men het gebruik van andere hotkeys gemakkelijk kan implementeren (bv dat men met “ctrl F2” naar zaterdag kan springen).

Opmerking: je mag geen gebruik maken van *KeyListener* of *KeyAdapter*.

4. Ik wil voor de verschillende dagen in de week een verschillende kleur verkrijgen en gebruik de volgende methode:

```
public void weekdagKleur (Weekdag dag) {  
    if ((dag == Weekdag.ZATERDAG) || (dag == Weekdag.ZONDAG)) {  
        return Color.GREEN;  
    } else {  
        if (dag == Weekdag.WOENSDAG) {  
            return Color.BLUE;  
        } else {  
            return Color.BLACK;  
        }  
    }  
}
```

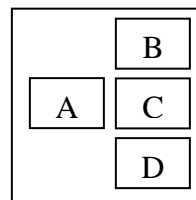
Schrijf op hoe je de if-else constructie kan vermijden als ...

\* je Weekdag niet mag aanpassen:

\* je Weekdag wel mag aanpassen:

5. We hebben een paneel waarin we de component willen schikken met *GroupLayout* op de manier zoals ze hier in de figuur beschreven staat.

Schrijf hier de schikking op zoals *GroupLayout* dat vereist. Je hoeft hier dus geen Java-code te schrijven, de schikking op zich is genoeg.



6. De interface *TableModel* bevat een methode *getColumnClass(int kolomIndex)*; waarmee je de meest specifieke klasse terugkrijgt waarvan de kolomelementen Objecten van zijn. Deze methode kan je zelf overschrijven om zo goed mogelijk een jouw tabel te beantwoorden.

1. Wat is het visuele voordeel dat je verkrijgt als deze *Integer.class* teruggeeft?
2. (Doordenkertje) Wat kan het voordeel zijn van hier *Weekdag.class* terug te geven?
3. Wat stelt '.class' hier eigenlijk voor?

7. Je hebt een database met daarin een tabel *prog2* die de velden *naam* (String) en *punten* (Integer) bevat. De connectie naar de database is reeds geopend. Geef een lijst van de studenten die minstens een gegeven aantal punten heeft. Eventuele foutmeldingen worden doorgegeven. De database-connectie moet open blijven!

```
public List<String> getStudenten(Connection conn, int punten) throws SQLException{  
}
```