

Übung 07: Klassen und Objekte

Abgabetermin: 08. 12. 2008

Name: _____

Matrikelnummer: _____

Gruppe: G1 (Prähofer)

G2 (Wolfinger)

G3 (Wolfinger)

G4 (Jahn)

Aufgabe	Punkte	gelöst	abzugeben schriftlich	abzugeben elektronisch	Korr.	Pkte
Aufgabe 07.1	12	<input type="checkbox"/>	Java-Programm Testergebnisse	Java-Programm	<input type="checkbox"/>	
Aufgabe 07.2	12	<input type="checkbox"/>	Java-Programm Testergebnisse	Java-Programm	<input type="checkbox"/>	

Aufgabe 07.1: Zeichenprogramm

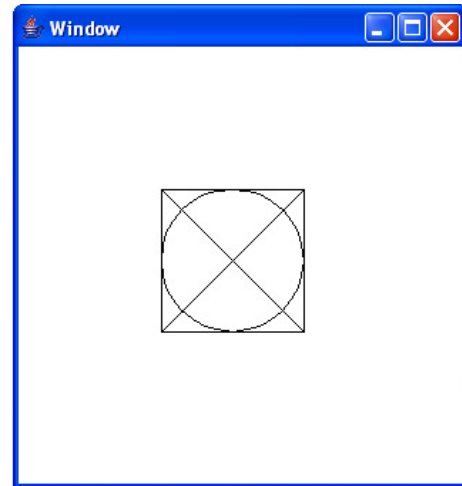
Die Klasse *Window.java* aus der Vorgabedatei zeichnet einfache graphische Objekte in ein Fenster. Die Methode *Window.open()* öffnet das Fenster und folgende Methoden zeichnen eine Linie, ein Rechteck oder einen Kreis:

- *drawLine*(int x1, int y1, int x2, int y2)
- *drawRectangle*(int x, int y, int w, int h)
- *drawCircle*(int x, int y, int r)

Implementieren Sie mit der Klasse *Window* ein einfaches Zeichenprogramm. Mit dem Zeichenprogramm soll man Linien (L=*Line*), Rechtecke (R=*Rectangle*) und Kreise (C=*Circle*) zeichnen können. Folgender Beispieldialog zeigt wie das Programm die Art der zu zeichnenden Figur und deren Koordinaten einliest. Die Eingabe eines Punktes beendet das Programm.

```
Figure (L|R|C): R
  Location (x, y): 100 100
  Width and Height (w, h): 100 100
Figure (L|R|C): C
  Center (x, y): 150 150
  Radius r: 50
Figure (L|R|C): L
  Point 1 (x1, y1): 100 100
  Point 2 (x2, y2): 200 200
Figure (L|R|C): L
  Point 1 (x1, y1): 100 200
  Point 2 (x2, y2): 200 100
Figure (L|R|C): .

All Figures:
- Rectangle (100,100) - 100 | 100
- Line (100,100) - (200,200)
- Line (100,200) - (200,100)
- Circle (150,150) / 50
```



Implementieren Sie Klassen für *Point*, *Line*, *Rectangle* und *Circle*. Eine eingelesene Figure soll unmittelbar nach der Eingabe in das Fenster gezeichnet werden. Zusätzlich sollen alle eingelesenen Figuren in Arrays vom Typ *Line[]*, *Rectangle[]* und *Circle[]* gespeichert werden. Geben Sie beim Programmende eine textuelle Beschreibung für alle Figuren auf der Konsole aus.

Hinweis: Die Klasse *Window.java* können Sie von der LVA-Homepage herunterladen.

Abzugeben sind: Java-Programm, Testergebnisse

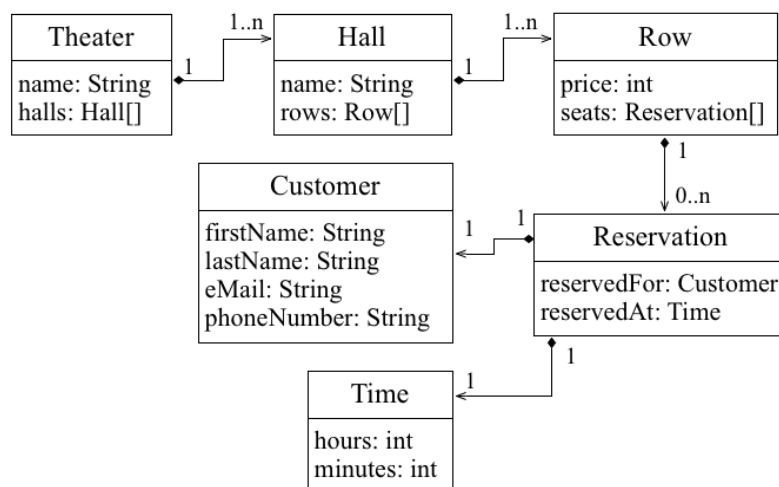
Aufgabe 07.2: Reservieren von Kinoplätzen

Peter Filmsky führt zwei kleine Programmkinos in oberösterreichischen Gemeinden. In den Kinos gibt es in jedem Saal täglich nur eine Vorstellung. Filmsky will modernisieren und seinen Kunden ermöglichen über Internet zu reservieren. Sie werden beauftragt ein Programm zu schreiben, das die eingelangten Reservierungen aus einer Eingabedatei liest und die Belegung in den Kinos zeigt.

Die nachstehende Abbildung zeigt die Ausgabe für Saal 1 eines Kinos: Saal 1 hat fünf Reihen. Es gibt Reihen mit 5 Sitzplätzen und Reihen mit 6 Sitzplätzen. Reservierte Plätze sind mit ** gekennzeichnet. Für freie Plätze wird der Preis einer Eintrittskarte gezeigt, zB Sitz 1 und 2 in Reihe 5 sind noch frei und kosten je 10 €.

Saal1	01	02	03	04	05	06
Reihe 1	5	5	**	**	5	
Reihe 2	5	5	5	5	5	
Reihe 3	7	7	7	7	7	7
Reihe 4	7	7	**	**	7	7
Reihe 5	10	10	**	**	**	

Implementieren Sie die Klassen *Theater*, *Hall*, *Row*, *Reservation*, *Customer* und *Time* wie im Klassendiagramm dargestellt. Ein Kino (*Theater*) hat einen Namen und besteht aus einem oder mehreren Sälen (*Hall*). Ein Saal hat einen Namen und besteht aus einer oder mehreren Sitzreihen (*Row*). Eine Sitzreihe hat einen Preis, der für jeden Platz in dieser Reihe gilt, und ein Array von Sitzplätzen. Die Länge des Array definiert wie viele Plätze es in der Reihe gibt. Steht an einer Stelle im Array der Wert *null*, bedeutet das, dass der Platz *frei* ist. Wenn der Platz reserviert ist, zeigt der Wert im Array auf ein Reservierungsobjekt (*Reservation*). Die Reservierung besteht aus Kundendaten (*Customer*) und einer Reservierungszeit (*Time*).



Implementieren Sie den Ablauf in der *main*-Methode der Klasse *Theater* wie folgt: Die Methode *readHalls* liest aus der Datei *halls.txt* wie viele Säle, mit wie vielen Reihen und Sitzen es gibt und erstellt die entsprechenden Objekte. Die Methode *readReservations* liest die eingegangenen Reservierungen aus der Datei *reservations.txt* und fügt die *Reservation*-Objekte ein.

```

public class Theater {
    static Hall[] halls;
    public static void main(String[] args) {
        readHalls("halls.txt");
        readReservations("reservations.txt");
        printHalls();
    }
    ...
}
  
```

Die Datei *halls.txt* aus der Vorgabedatei hat folgenden Aufbau: In der ersten Zeile steht wie viele Säle die Datei enthält. Jede weitere Zeile beschreibt einen Saal mit Name, Anzahl der Reihen, und Anzahl der Sitze/Preis pro Eintrittskarte.

```
2
Saal1 7 10/8 10/8 12/8 12/9 14/9 14/10 15/6
Saal2 5 10/6 10/7 12/8 10/8 12/9
```

Die Datei *reservations.txt* aus der Vorgabedatei hat folgenden Aufbau: Jede Zeile enthält eine Reservierung beginnend mit Name des Saals und Zeit der Reservierung. Darauf folgt der Kunde mit Vorname, Nachname, EMail-Adresse und Telefonnummer. Am Zeilenende folgt die Anzahl der reservierten Plätze und für jeden Platz Reihe/Platz.

```
"Saal1" 10:30 "Babett" "Ahrens" "holger@web.de" "+43 (699) 9966026" 2 1/3 1/4
"Saal1" 9:50 "Barbara" "Albert" "gallileo@gmx.at" "+43 (664) 9966011" 3 5/3 5/4 5/5
"Saal1" 17:10 "Nora" "Aschacher" "jopy@yahoo.de" "+43 (676) 9966040" 2 4/3 4/4
```

Hinweise:

- Sie können davon ausgehen dass die Dateien ausschließlich korrekte Daten enthalten.

Abzugeben ist:

- Java-Programm
- Testergebnisse