

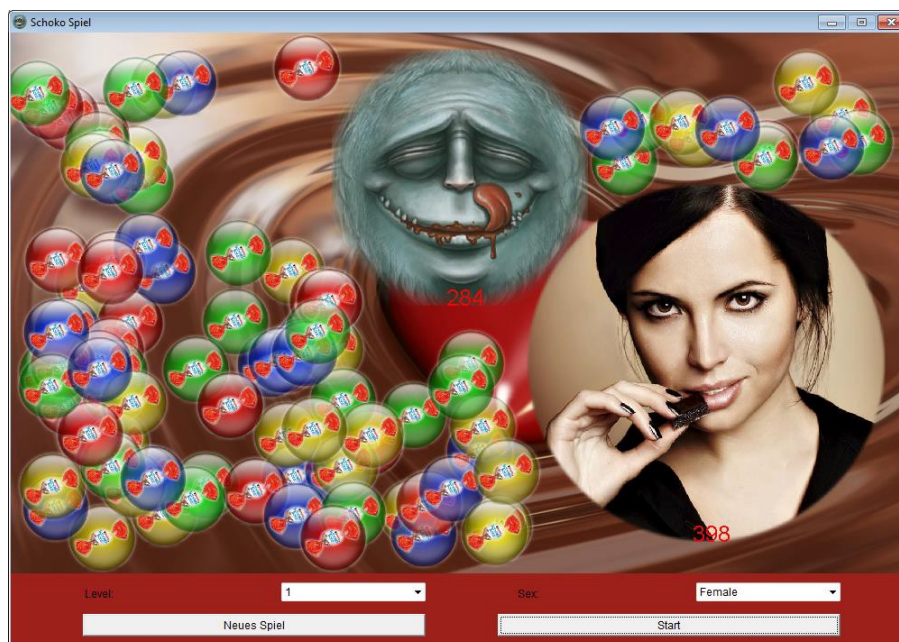
# JAVA PRÜFUNG I

## Bedingungen:

- Erlaubte Hilfsmittel: Unterrichtsunterlagen, Java Buch und Übungen.
- Die Prüfung ist schrittweise, gemäss Aufgabenstellung lokal auf Ihrem Computer zu lösen. Kopieren sie zu diesem Zwecke den gesamten Ordner *P1\_HS15\_16\_ChocolateMonster\_Vorlage* auf Ihre lokale Harddisk und importieren sie das Projekt in Eclipse. Am Ende der Prüfung ist der Ordner *src* umbenannt in *NameVorname* abzugeben.
- Setzen sie als erstes Ihren Namen und Vornamen in die Dateien.
- Gegenseitiges Abschreiben in irgendeiner Form führt zur Note 1!
- Folgend sie bei der Wahl von Variablen exakt den Angaben in der Aufgabenstellung.
- Die Beilage mit dem Layout muss unterschrieben zurückgegeben werden!

## Beschreibung:

Ziel dieser Prüfung ist es, die Spiel-Applikation **Schoko Spiel** zu programmieren. Das Spiel ist ein Schoko-Wettessen gegen ein Monster: Die Schokolade-Kugeln und das Monster bewegen sich zufällig über den Bildschirm, die Person folgt der Maus. Jede Kugel, die mit dem Monster oder der Person in Berührung kommt wird aufgegessen. Kommen Monster und Person in Berührung endet das Spiel. Wer die Grössere Zahl Schoko-Kugeln verzehrt hat, hat gewonnen. Beim Spiel lassen sich Spiel-Level und Art der Person wählen. Mittels Button *Neues Spiel* lässt sich ein neues Spiel erzeugen, mit Button *Start* wird das Spiel gestartet. Die Maus dient dazu, die Person zu steuern.



Die Applikation ist im null – Layout erstellt und das Layout ist gemäss Angaben in der Beilage organisiert. Die Benutzerschnittstelle sowie das Schoko-Spiel befinden sich auf der *TopView*. Das *Schoko-SpielPanel* besitzt einer *SimpleTimer* um die notwendigen Updates durchzuführen. Das Panel hat weiter die Person, das Monster und einen Array mit den Schoko-Kugeln. Alle Drei sind von der Art *Figur*. Das *SchokoSpielPanel* hat ein Hintergrundbild, die Figur verfügt über das entsprechende Bild. Die detaillierten Angaben sind dem Klassendiagramm in der Beilage zu entnehmen.

Die Aufgabenstellung führt Sie schrittweise zum Ziel. Folgen Sie daher beim Lösen der Prüfung der Aufgabenstellung. Sämtliche Bilder sind im Ordner src/bilder zu finden und selbstsprechend bezeichnet.

**Achten Sie darauf, dass Ihr Code kompilierbar bleibt!**

### **Aufgabe 1: Erstellen des Grundgerüsts (~5 Pte.)**

Als erstes wollen wir das Grundgerüst des Programms anhand des Klassendiagramms kontrollieren und allenfalls ergänzen. Viele Elemente der Klassen sind dabei bereits gegeben.

- a) Ergänzen sie sämtliche *Attribut*- und *Methodendeklarationen* gemäss Klassendiagramm. Machen Sie sich dabei mit den vorhandenen Attributen und Methoden-Deklarationen bekannt.

### **Aufgabe 2: Klasse *TopView*: User - Interface (~29 Pte.)**

Als erstes wollen wir das User – Interface sowie den *ActionListener* und den *ItemListener* implementieren. Die Anordnung der Elemente geschieht im null – Layout gemäss Beilage.

- a) Kompletтировать Sie die Klasse *TopView* gemäss Dokumentation und Beilage.

Beispiel zur Komponente JComboBox:

```
auswahlLand = new JComboBox<String>();  
auswahlLand.addItem("Schweiz");  
auswahlLand.addItem("Deutschland");
```

### **Aufgabe 3: Klasse *Figur* (~31 Pte.)**

Als nächstes wollen wir uns der Klasse *Figur* zuwenden. Achten Sie beim Laden des Bildes darauf, dass Sie *Utility.loadResourceImage()* verwenden!

- a) Implementieren Sie die Klasse *Figur* gemäss Dokumentation. Laden Sie ein bestimmtes Schoko-Kugel-Bild, wenn Ihnen die Sache mit dem Zufall nicht gelingt!

Testen können Sie die Klasse, indem Sie in der Klasse *SchokoSpielPanel* eine *Figur* erzeugen:

```
Figur figur = new Figur(0);
```

und in *paintComponent()* zur Anzeige bringen.

### **Aufgabe 4: Klasse *SchokoSpielPanel* (~39 Pte.)**

- a) Die Klasse *SchokoSpielPanel* beheimatet die *Schoko-Kugeln*, das *Monster* und die *Person*, sowie die zum Spielen notwendigen Methoden.
- b) Implementieren Sie den Konstruktor und die Methoden der Klasse *SchokoSpielPanel* gemäss Dokumentation.
- c) **Ergänzen Sie den Code um noch allfällig fehlenden Code.**