

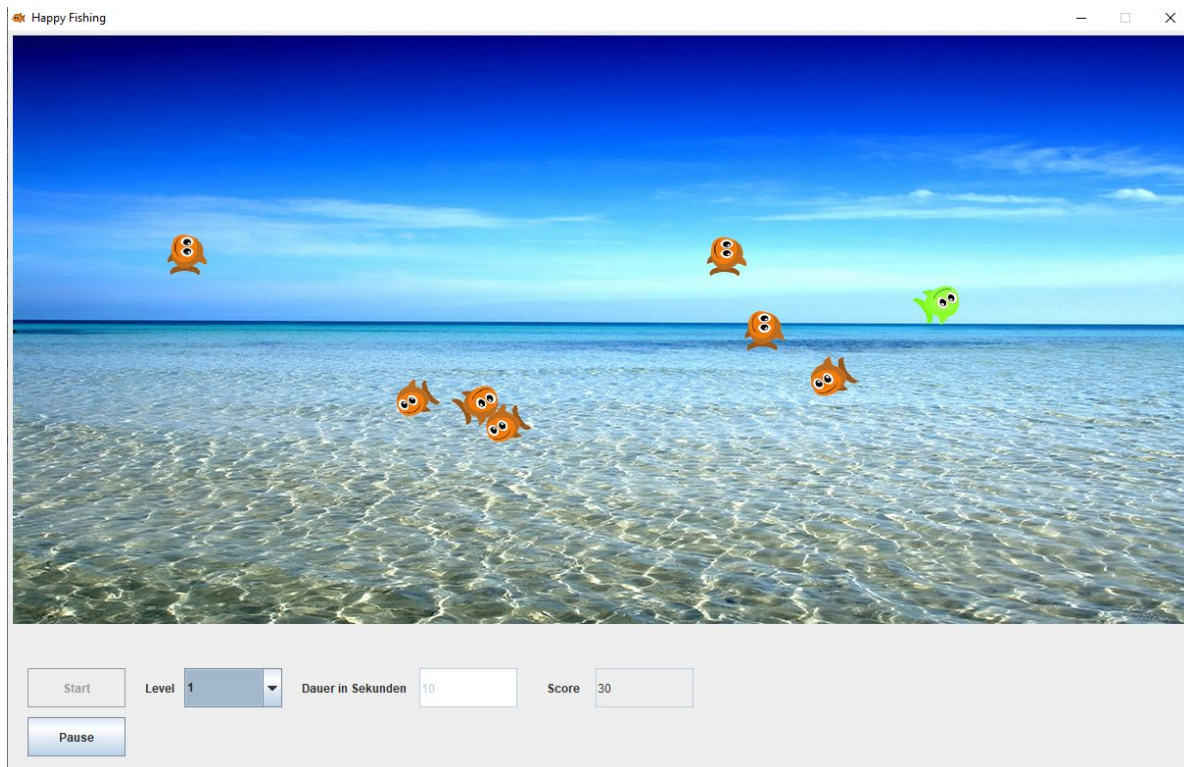
JAVA PRÜFUNG

Bedingungen:

- Erlaubte Hilfsmittel: Unterrichtsunterlagen, Java Buch und Übungen.
- Alte Prüfung, soweit nicht als Übung abgegeben, sind nicht erlaubt!
- Die Prüfung ist schrittweise, gemäss Aufgabenstellung lokal auf Ihrem Computer zu lösen. Kopieren sie zu diesem Zwecke den gesamten Ordner *P_HS20_21_HappyFishing_Vorlage* auf Ihre lokale Harddisk und importieren sie das Projekt in Eclipse.
- Am Ende der Prüfung ist der Ordner *src* umbenannt in *NameVorname* gezippt als per Email 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 und der User-Interface Positionierung exakt den Angaben in der Aufgabenstellung.

Beschreibung:

Ziel dieser Prüfung ist es, die Spiel-Applikation *HappyFishing* zu programmieren. Im Spiel springen vergnügte Fische vor traumhafter Kulisse. Ziel des Spiels ist es, per Mausklick so viele Fische wie möglich zu fangen. Gelegentlich erscheinen grüne kostbare Fische, die besonders lohnend zu angeln sind.



Die Applikation ist im null-Layout erstellt und das Layout ist gemäss Angaben in der Beilage organisiert. Die Benutzerschnittstelle befindet sich auf dem Spielpanel. Die Spieldauer wird im Textfeld vorgegeben. Mittels einer JComboBox können drei Level vorgegeben werden, was die Anzeigedauer

der Fische beeinflusst. Mit dem Pause Button wird das Spiel unterbrochen und wieder fortgesetzt. Nach Ablauf der Spielzeit erscheint das GameOver Bild auf dem **SpielPanel**. Die detaillierten Angaben zur Programmstruktur 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. Laden Sie alle Bilder mit **Utility.loadImage(String strBild, int width, int height)**. Die Breiten und Höhen sind im Code angegeben.

Achten Sie darauf, dass Ihr Code kompilierbar bleibt!

Viel Erfolg!

Aufgabe 1: Erstellen des Grundgerüsts(~ ? Punkte)

Als erstes wollen wir das Grundgerüst des Programms anhand des Klassendiagramms nachvollziehen und uns mit dem Aufbau des Programms vertraut machen. Attribute und Methoden sind alle vorhanden und müssen nicht detailliert kontrolliert werden.

Aufgabe 2: Klasse SpielPanel: User-Interface (~ 21 Punkte)

Nun realisieren wir das User- Interface des Spielpanels gemäss Beilage. Implementieren Sie hierzu den **Konstruktor** und die **paintComponent** Methode gemäss Dokumentation. Danach können Sie bereits den herrlichen Ausblick geniessen.

Die Auswahlliste für die **JComboBox** ist im String Array **levelList** abgelegt

Aufgabe 3: Klasse GraphikObjekt (~ 8 Punkte)

Als nächstes kümmern wir uns um die Klasse **GraphikObjekt**, welche an Klasse **Fisch** vererbt. Implementieren Sie die Klasse **GraphikObjekt** gemäss Dokumentation im Code.

Aufgabe 4: Klasse Fisch (~ 20Punkte)

Um die Fische lebendig werden zu lassen implementieren wir als nächstes den **Konstruktor** und die **paint** Methode der Klasse **Fisch**. Um die Fische zu rotieren, verwenden Sie zur Erzeugung des Bildes die vorgegebene Methode **rotateImage**, die Sie in der Klasse bereits vorfinden.

Aufgabe 5: Klasse SpielPanel Event und Timer Programmierung (~ 30 Punkte)

Implementieren Sie nun die **actionPerformed** und die **timerAction** Methode der Klasse **SpielPanel** entsprechend der Dokumentation im Code. Wenn Sie das Spiel nun starten sollten für die eingetragene Spieldauer fröhliche Fische durch die Wasserlandschaft springen.

Aufgabe 6: Klasse SpielPanel und Klasse Fisch vervollständigen (~ 25 Punkte)

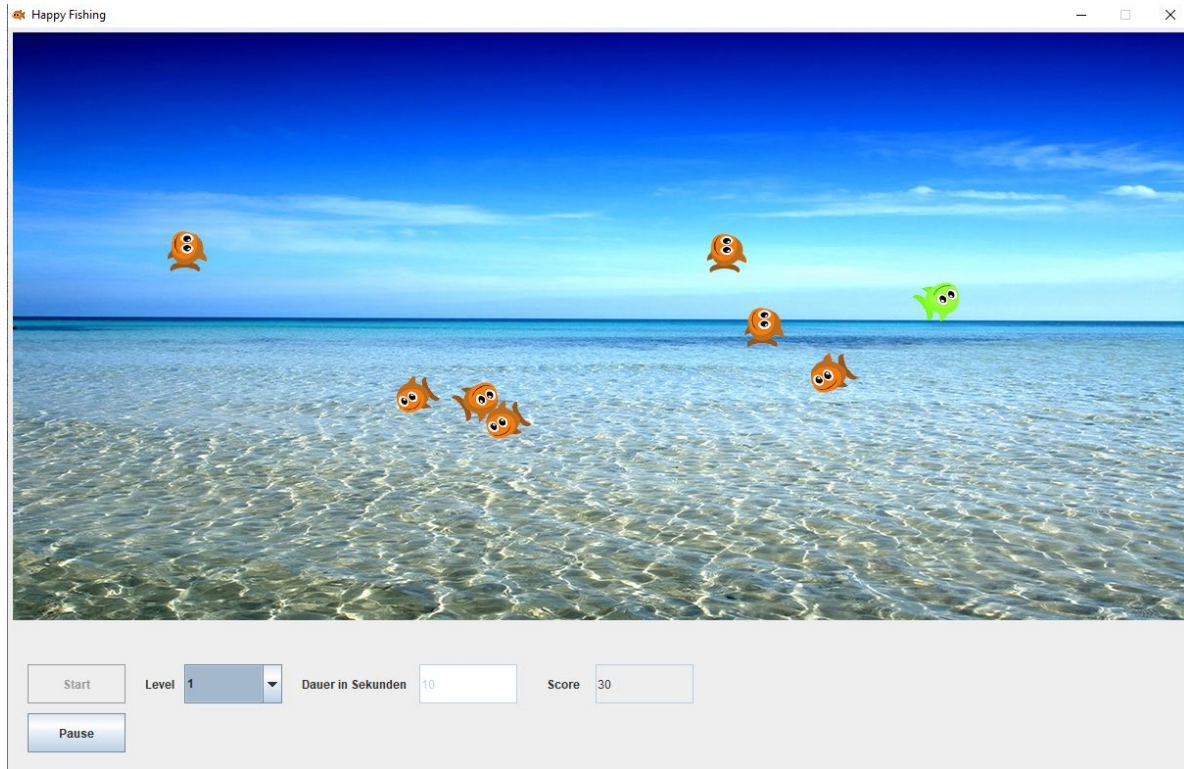
Nun wollen wir noch mit der Maus die Fische fangen und das Spiel durch unterschiedliche Levels schwieriger machen. Vervollständigen Sie die Klassen **SpielPanel** und **Fisch** gemäss der Doku.

Nun können Sie Angeln gehen. Die grünen Fische zählen doppelt!

Petri Heil!

JAVA PRÜFUNG

Layout:



Element	x – Position	y – Position	Breite	Höhe
<i>JButton</i> "Start"	20	650	100	40
<i>JLabel</i> "Level"	140	650	100	40
<i>JComboBox</i> für Level	180	650	100	40
<i>JLabel</i> "Dauer in Sekunden"	300	650	150	40
<i>JTextField</i> Dauer	470	650	100	40
<i>JLabel</i> "Score"	600	650	100	40
<i>JTextField</i> Score	650	650	100	40
<i>JButton</i> "Pause"	20	700	100	40

Die Elemente müssen exakt so positioniert werden!

Ich bestätige, dass ich die Prüfung selbständig und ohne fremde Hilfe gelöst habe

Windisch, 13.01.2021

Ort / Datum

Name/Vorname

Unterschrift

Klassendiagramm:

