

Spielregeln

Für das Praktikum B-TI2 PRP2 gelten in diesem Semester die folgenden *Spielregeln*:

1. Tragen Sie sich im StiSys in die richtige Praktikumsgruppe ein, falls Sie dort noch nicht in der richtigen eingetragen sind!
2. Bearbeiten Sie die Aufgaben bitte in Zweierteams!
3. Geben Sie Ihrem Team bitte einen Namen! Unter diesem veröffentliche die Zwischenergebnisse.
4. Ich erwarte Ihre Programmier-Lösungen in einer Form, in der sie unter Eclipse in Java laufen. Ein Projekt pro Team, ein Paket (package) pro Aufgabe, ggf. weitere Teilpakete. Ich werde bei der ersten Aufgabe ein neues Projekt pro Team anlegen, auf Basis der von Ihnen geschickten .zip Datei (s.u.). Bei den weiteren Aufgaben erwarte ich, dass nach einem import→archive Alles läuft.
5. Zu den meisten Lösungen wird ein Text mit Erläuterungen, Begründungen o.ä. gehören. Diesen akzeptiere ich in jedem Format dass ich mit Open Office bearbeiten kann oder in L^AT_EX bzw. T_EX. Zu Ihrer eigenen Sicherheit empfehle ich zusätzlich eine PDF-Version abzugeben.
6. Alle Bestandteile ihrer Lösung (Java-Sourcen, Text, ...) fügen Sie bitte zu einer zip-Datei mit dem Namen **ann.zip** hinzu.
7. Lösungen schicken Sie bitte per email an

bernd.kahlbrandt@informatik.haw-hamburg.de.
8. In die Betreffzeile Ihrer Email schreiben Sie bitte: **PRnnTeamName**, wobei *nn* die Nummer der Aufgabe und *TeamName* Ihr Teamname ist.
9. Die Datei mit Ihren Lösungsbestandteilen füge Sie bitte als Anlage dieser Email bei.
10. Fordern Sie bitte keine Empfangsbestätigung an. Ich werte das als Nötigung!

11. Geben Sie bitte alle verwendeten Quellen in professioneller Weise¹ an! Das gilt selbstverständlich auch für Quellen aus dem Internet!
12. In jedes Dokument gehören die Namen der Autoren!
13. Achten Sie bitte darauf, dass stets alle Bestandteile dabei sind, die ich benötige, um die Lösung auszuprobieren und zu beurteilen. Wenn ich „Ihr“ Projekt angelegt habe, so erwarte ich Folgendes:
 - 13.1. Ich kann Ihre zip-Datei in Eclipse importieren. Es gibt keine Warnungen. Die Lösung lässt sich ausführen und die Testfälle werden bestanden.
 - 13.2. Erläuterungen akzeptiere ich in einem Format, das OpenOffice bearbeiten kann oder in L^AT_EX bzw. T_EX. Damit Sie die Kontrolle über das Erscheinungsbild haben, empfehle ich auch eine pdf-Version des Dokuments abzugeben.
 - 13.3. Eine Lösung, in der die Java-Direktive „@SuppressWarnings“ vorkommt, ist in der Regel nicht akzeptabel! Einige Ausnahmen können Sie aber begründen.
14. Prognosen sind schwierig, besonders wenn sie die Zukunft betreffen, aber es wird voraussichtlich jede Woche ein Aufgabenblatt geben.
15. Ist nichts anderes angegeben, so ist der späteste akzeptable Abgabetermin für die Aufgaben am **Mittwoch nach Ausgabe, 12:30**, Eingang in meiner Mailbox.
16. Eine Aufgabe wird als akzeptabel gelöst akzeptiert, wenn folgende Kriterien (Minimalanforderungen) erfüllt sind:
 - Die Lösung geht termingerecht bei mir ein.
 - Die Lösung genügt den formalen Kriterien, so dass ich sie einfach in Eclipse importieren kann.
 - Der Code liefert das spezifizierte Ergebnis. Bei einigen Aufgaben werde ich mit der Ausgabe der Aufgabe JUnit Testfälle zur Verfügung stellen. Diese müssen dann bestanden worden sein!
 - Der Code genügt den in der Vorlesung präsentierten Qualitätskriterien. Dazu gehören in diesem Semester mindestens folgende Dinge:
 - Die Java Code-Konventionen werden eingehalten!
 - Jede Klasse enthält einen Javadoc Kurzkomentar und die Namen der Autoren (@author tag). Weitere Javadoc-Kommentare können Sie schreiben. Sie werden sie schreiben müssen, sobald Javadoc behandelt worden ist. Ich verweise aber bereits jetzt auf das entsprechende Kapitel im Skript!
 - Es gibt keine Code-Redundanzen.
 - Entwurfsentscheidungen sind überzeugend begründet.

¹Siehe z. B. users.informatik.haw-hamburg.de/home/pub/prof/kahlbrandt/gen/litverz.pdf

- Sie sind über den Code auskunftsfähig.

17. Jede(r) Teilnehmer(in) muss Lösungen im Praktikumstermin hinreichend häufig (alle gleich oft) präsentieren und die Fragen der anderen Teilnehmer befriedigend beantworten.

Hier nun eine Beispielaufgabe, für die ich einen Lösungsvorschlag in meinem pub zur Verfügung stelle.

Aufgabenblatt 0

In diesem Aufgabenblatt sollen Sie den Umgang mit Swing Componenten trainieren.

1. Verwenden Sie bitte eine der Zählerklassen (Counter, Counternn) aus dem letzten Sommersemester!
2. Schreiben Sie bitte in Java eine Unterklasse von JPanel, die ein Zählerobjekt darstellt und die Interaktion ermöglicht! Der Zähler soll mit einem Anfangswert initialisiert werden können, erhöht, verringert und auf den Anfangswert zurückgesetzt werden können.

Auf gute Zusammenarbeit und viel Erfolg!