

Aufgaben – Teil 1

Vorbereitende Aufgaben

1. Recherchieren Sie die Funktionsweise der Klasse *StringBuilder*
 - 1.1. Implementieren Sie eine Applikation, die verschiedene Variablen initialisiert und mittels *StringBuilder* in eine einzelne Ausgabe kombiniert
 - 1.2. Schreiben Sie eine Klasse *Bananentransportkiste*, die verschiedene Attribute besitzt. Erstellen Sie ein neues Objekt dieser Klasse und bauen Sie es in die Ausgabe mit ein
2. Erarbeiten Sie die Unterschiede der Methoden *String.format()* und *String.formatted()*
3. Reproduzieren Sie das Beispiel von Aufgabe 1 mittels *String.format()* oder *String.formatted()*
4. Nutzen Sie einen *java.util.Scanner* und den InputStream *System.in*, um mittels der Methode *nextLine()* Nutzereingaben zu lesen
 1. Nutzen Sie einen *switch case*-Block, um je nach Eingabe eine andere Ausgabe zu erzeugen

Prüfungsrelevante Aufgaben

1. Erstellen Sie ein neues Java-Projekt und fügen Sie die Spielsimulation als .jar-Datei Ihrem Projekt bei, konfigurieren Sie diese als Abhängigkeit
2. Prüfen Sie, dass Sie in Ihrem Projekt Klassen aus dem Paket *gmbh.kdb.hsw.gdp* nutzen können, indem Sie beispielsweise die statische Methode *gmbh.kdb.hsw.gdp.Game#create(IGameHandler)* mit *null* aufrufen und das Programm kompilieren
3. Ermitteln Sie die Funktionsweise der Methode *gmbh.kdb.hsw.gdp.Game#create(IGameHandler)* und erstellen Sie eine neue Spielinstanz, die das Spiel nach der ersten Runde beendet
4. Implementieren Sie eine Klasse *MainMenu*, die schließlich das Hauptmenü der Applikation ausgeben soll
 1. Bauen Sie eine Menüstruktur als Textausgabe und implementieren Sie zwei Möglichkeiten:
 1. Beenden des Spiels
 2. Fortsetzen mit der nächsten Runde

Beachten Sie dazu jeweils die Rückgabewerte von
IGameHandler#run(GameDevStudio)

2. Lesen Sie Eingaben von der Konsole mittels *Scanner* ein und führen Sie die gewählte Aktion aus
3. Lassen Sie das Programm jeweils so lange fortfahren, bis das Spiel mit einer Fehlermeldung bzgl. Konkurs abbricht

Hinweis: Es wird empfohlen, nur eine *Scanner*-Instanz zu verwenden. Führen Sie daher Eingaben nur an einem Punkt durch oder geben Sie den Scanner jeweils an andere Klassen weiter oder verwenden Sie das [Singleton-Pattern](#).