LI - Arbeit 03  
Objekt Orientierte Programmierung mit Python

Dokumentieren Sie jeweils passend den Quellcode aus Visual Studio Code (o.ä.) als Text fest (mit Kommentaren und dem Eigenen Namen).   
Halten Sie die Ausgaben des Programms als Bild/Bilder fest! Auch ggf. zugehörige Informationen wie die Datenbank Tabelle, Bild für Panda o.ä. ergänzend fest. Backen Sie den Gesamten Quellecode in ein ZIP mit „NAME\_VORNAME\_LK21“. Geben Sie das Zip mit „allem“ und die Dokumentation ab!

Hilfsmittel: W3CSchools, Visual Studio Code Hilfe und Tools, Quellcode und Dokumentation MySQL Workbench/PHPMyAdmin   
**Keine zusätzlichen Hilfsmittel – kein Internet! Zeit: 135 min**

Bitte beachten Sie:

Es wird nur das gewertet, was auf den Aufgabenblättern steht, Dokumentiert und per Digital abgeliefert wurde! Zusätzliche Blätter dürfen **nicht** benutzt werden (nur nach individueller Rücksprache). Sollte der Platz nicht ausreichen, schreiben Sie bitte auf die Rückseite und **kennzeichnen** dies deutlich. Schreiben Sie mit Kugelschreiber oder Füllfederhalter. Verwenden Sie keinen Rotstift. **Schreiben Sie auf jedes Blatt und in jede Datei als Kommentar Ihren Namen! Jeder Austausch untereinander (auch digital) wird als Täuschungsversuch gewertet!** **Speichern Sie regelmäßig auf HDD und USB!**

|  |  |  |
| --- | --- | --- |
| Aufgabebereich | Punkte | Erreichte Punkte |
| 1.1 | 19 |  |
| 1.2 | 14 |  |
| 2.1 | 20 |  |
| 3.1 | 20 |  |
| 4.1 | 20 |  |
| 5.1 | 17 |  |
| Gesamt | **110** |  |

Ihre Note:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Bewertet von:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Basics Python 19 + 14

Erstellen Sie eine Klasse

* **HooniganTuning** sowie
* **Auto**

## Die Klasse HooniganTuning

Beim Erstellen eines Objekts z.B. myHooniganTuning der Klasse HooniganTuning wird in der \_\_init\_\_ Methode auf dem Terminal der Benutzer mit „Welcome to HooniganTuning!“ begrüßt und danach werden von Ihm die folgenden Werte erfragt:

* Marke (Text)
* Modell (Text)
* PS (Zahl)
* Farbe (Text)
* Elektro (True oder False)

Anschließend wird damit ein Auto erstellt, dieses durch Aufruf von „tuning()“ getuned und die gesamten Daten auf dem Terminal ausgegeben.

## Die Klasse Auto

* hat die Felder *marke, modell, ps, farbe und elektro*. Beim Erstellen eines Objekts aus der Klasse müssen diese Werte übergeben werden und werden dann in der \_\_init\_\_ auf die Felder/Variablen des Objekts gesetzt.
* Die Methode „tuning“ der Klasse Auto erhöht den Wert von PS um 15% (also mal 1,15) und setze diesen nun als Wert auf das Feld ps, also überschreibt diesen (aus 100 werden dann 115 PS).
* Durch das Überschreiben der \_\_str\_\_ Methode wird in der Klasse Auto dafür gesorgt das alle Werte als Text ausgegeben werden. Also z.B. so: „Marke: Ford; Modell: FiestaRS; PS: 275; Farbe: Blau; Elektor: True“

## Lösung:

Tabelle 1 Quellcode Basics zum Aufgaben Teil HooniganTuning

|  |
| --- |
|  |

Tabelle 2 Quellcode Basics zum Aufgaben Teil Auto

|  |
| --- |
|  |

# QT5 20

Erstellen Sie einen Dialog in der Klasse **FriendlyPython**:

* in dem Sie In einem QLineEdit eine Zahl eintragen können
* in dem Sie in einem QLineEdit einen Namen eintragen können
* geben Sie QLabels für die QLinesEdits an mit „Anzahl“ und „Namen“
* erstellen Sie einen Button „Begrüßen“
* Beim Klick des Buttons wird entsprechend er Zahl mal „Hallo “+Name im Terminal ausgegeben.

## Lösung

Tabelle 3 Quellcode zur Aufgabe QT5

|  |
| --- |
|  |

# Panda 3D 20

Erstellen Sie eine Klasse **MyLittlePanda** in der Sie:

* Erstellen Sie ein Panda3D Fenster mit dem Titel LKArbeitvonNAME2021.
* Setzen Sie die Auflösung auf 800x600 Pixel.
* Laden Sie ein Model (z.B. „plane.egg“) und setzen sie es mittel reparentTo auf die camera.
* Erstellen Sie eine PNG oder JPEG als Textur von 800x600 mit Paint das einfach Blau ist.
* Laden Sie dies als Texture für das Model.
* Zeigen Sie nun dieses Model mit Texture an.

## Lösung:

Tabelle 4 Quellcode zur Aufgabe Panda3D

|  |
| --- |
|  |

# Datenbanken 20

Zeigen Sie das Sie mit Python und einer MySQL/MariaDB kommunizieren und Daten abfragen können!

* Bauen Sie eine Verbindung zu Ihrer Datenbank in der Klasse **LittleDBHelper** auf!
* Tragen Sie dort den Wert „LKArbeitvonNAME2021“ in ein Beliebiges Feld ein (z.B. Name) (Anmerkung NAME ist hierbei Ihr Name z.B. LKArbeitvonMarcR2021).
* Fragen Sie die Daten aus der Tabelle ab und geben Sie diese im Terminal aus (print), so das auch LKArbeitvonNAME2021 erscheint plus andere Zeilen erscheinen.

## Lösung:

Tabelle 5 Quellcode zur Aufgabe Datenbank

|  |
| --- |
|  |

# BCrypt 17

Erstellen Sie eine extra Klasse **HashBrothers**, welche mit Hilfe von BCrypt überprüft welches der folgenden Passwörter (siehe unten) Korrekt ist im Bezug zu dem folgenden Hash Code:

**$2b$12$acCJTPaaIx56B9eG0nv9HOHp8zaC7PDlingzE68LyzUPSSfzv3sW6**

Der Hash („Bitte Hashcode eingeben: “) und das zu prüfende Passwort („Bitte Passwort eingeben: “) sollen durch Input vom Benutzer eingegeben werden können. Anschließend Erscheint „Passt“ wenn das Passwort Korrekt ist oder „Passt Nicht!“ wenn es nicht zum Hash passt.  
Halten Sie Code und Ausgaben des Programms fest. Prüfen Sie mindestens die Passwörter:

* HansPeter
* Han5P3t3r
* Hansipit

## Lösung:

Tabelle 6 Quellcode zur Aufgabe BCrypt

|  |
| --- |
|  |