

# Organisatorisches

---

**Zertifikatskurs Python**

# Wer sind wir und warum sind wir hier?

---

- **Ben Stock**

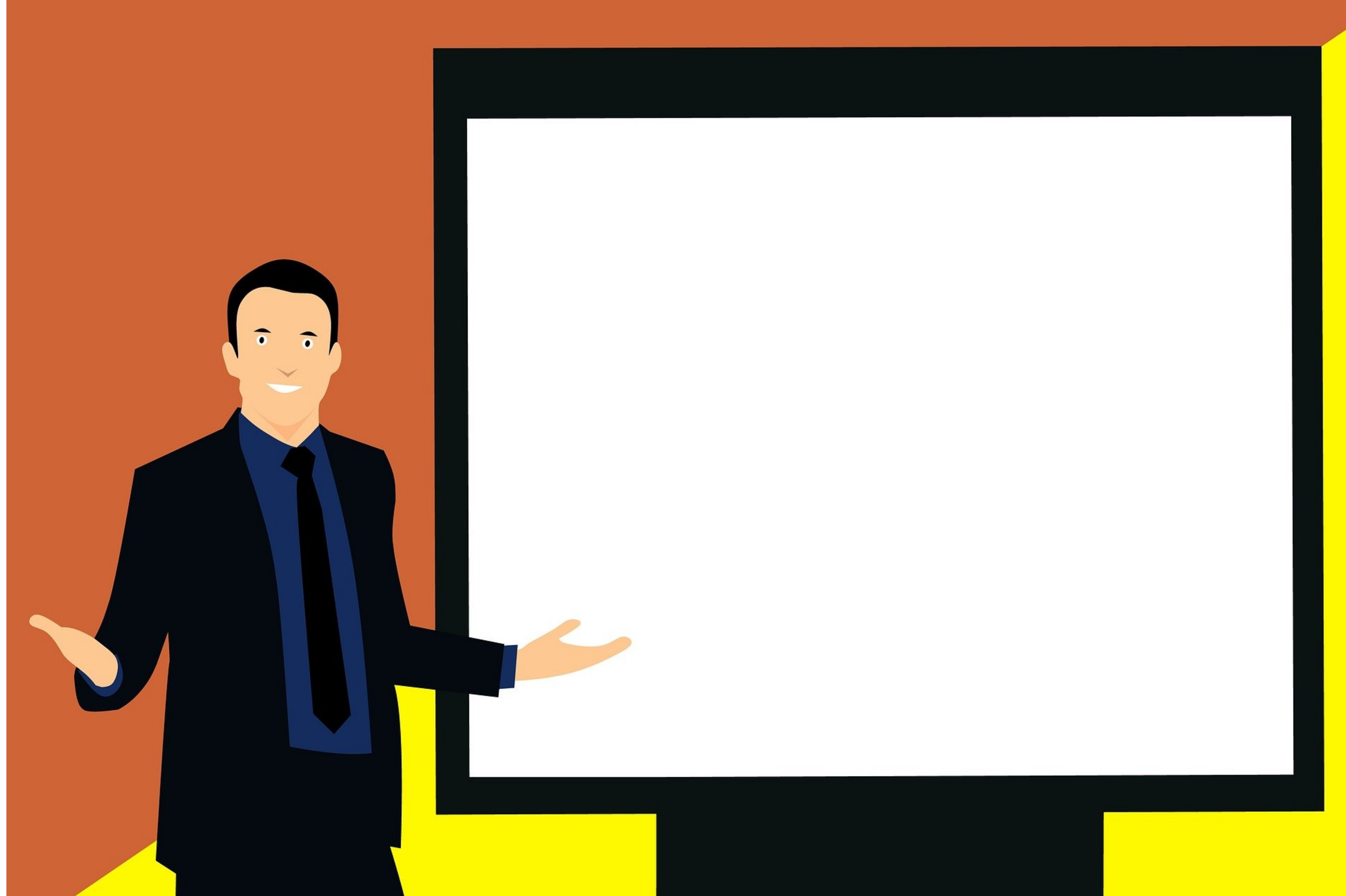


- Forschungsgruppenleiter am CISP A Helmholtz-Zentrum für Informationssicherheit
- Dozent für Web Application Security im CSP-Programm

- **Simon Einzinger**



- Studentischer Mitarbeiter am CISP A
- Teaching Assistant für Foundations of Cybersecurity 1 (inkl. Pythonkurs)



## Kurze Vorstellungsrunde

- Was macht ihr in eurem Team?
- Kommt ihr mit Python in Kontakt?
- Falls noch nicht, kommt ihr mit anderen Programmiersprachen in Kontakt?

# Lernziele und Lernart im Modules

---

- Lernziel: Python!
- Lernart: Praktisch
- Live-Meetings per Zoom (werden aufgezeichnet) inkl. Coding-Tasks in der Gruppe
- Anschließend Übungsaufgaben - jeweils **einzeln**
  - **Benotung basierend auf den erfolgreich gelösten Aufgaben**
  - Automatisiertes System zum Testen der Abgaben inkl. Feedback

# Organisation - Termine

Datum	Themen	Challenges
24.11.2025	Grundlegende Syntax, Einfache Datentypen, Arithmetic, Funktionen	10
1.12.2025	Kontrollfluss (Schleifen etc), erweiterte Datenstrukturen	10
15.12.2025	Comprehension, File Handling, Exceptions, Importing	10
5.1.2026	Objektorientierung in Python, Requests	10
22./23.1.2026	<b>Deadline für ersten 20 Tasks</b> <b>Präsenz-Tage:</b> Reguläre Ausdrücke, Security (SQLi + CMDi)	16 (6 optional)
4.2.2026	<b>Deadline Abgaben aller Übungsaufgaben</b>	-

# Organisation - Präsenzwochenende (tatsächliche Planung)

22.1.2026		23.1.2026	
10:30 - 10:45	Begrüßung	9:00 - 10:00	<b>Web Security: CMDi</b> (Aufgaben optional)
10:45 - 11:45	<b>Regular Expression</b>	9:00 - 10:00	<b>Tasks + "Office Hour"</b>
11:45 - 12:45	Mittagessen	10:00 - 10:15	Pause
12:45 - 14:00	<b>Tasks + "Office Hour"</b>	10:15 - 12:00	<b>Tasks + "Office Hour"</b>
14:00 - 14:30	Pause	12:00 - 13:00	Mittagessen + Feedback-Runde
14:30 - 15:30	<b>Web Security: SQL Injections</b>		
15:30 - 17:00	<b>Tasks + "Office Hour"</b>		

# Organisatorisches

---

- Vierte Iteration des Python-Kurses
  - Erfahrung auf fünf Iterationen Web Application Security und drei Python zeigt: Deadlines helfen, Stress am Ende vom Kurs zu reduzieren
- Arbeiten in Gruppen am Präsenz-Termin / in Meetings erwünscht
  - restliche Laufzeit: gerne austauschen, aber keine Lösungen kopieren
- Kommunikation mit uns: am besten per Mattermost
  - Zugangsdaten fürs Gitlab als SSO für Mattermost (und Gameserver)

# Feedback aus vorherigen Jahren und Änderungen

---

- Generell sehr positives Feedback aus dem letzten Jahr
  - Praxisnah, Hands-On, keine extra Prüfung
- Einige “Macken”
  - Aufgabenstellungen teilweise etwas unklar => verbessert, gerne **fragen!**
  - Aufgaben waren nicht gleichmäßig verteilt => weniger Stoff pro Termin, nur 10 Tasks pro Meeting
  - Eine sehr schwierige Aufgabe ohne genauere Beschreibung => bleibt so :-)



# Python - was ist das?

---

- Python ist eine interpretierte Programmiersprache
  - streng genommen: wird auf Bytecode kompiliert, danach ausgeführt (ähnlich zu Java)
- Besonderheiten: Ducktyping ("If it walks like a duck and it quacks like a duck, it's a duck")
  - keine feste Typendefinition bei "Deklaration" einer Variablen
- "Python comes with batteries"
  - große Standardbibliothek für viele Zwecke (HTTP-Verbindungen, Sockets, Reguläre Ausdrücke, Datei-/Ordnerverwaltung, ...)

# Python - was ist das?

---

- Riesiges Ökosystem an Dritt-Modulen
  - Python Package Index mit 450,000+ Modulen
  - einfach zu installieren über "pip" oder "easy\_install"
    - `pip install <package>`
    - pip-Suche derzeit ausgeschaltet, Paketname muss bekannt sein
- Aktuelle Version (Stand 13.11.2024): 3.13
  - Python 3 seit 2008
  - Python 2 seit 2000, end-of-life in 2020
  - Syntax leicht unterschiedlich, daher nicht kompatibel zueinander

# Python - wie code ich?

---

- Python rein interpretierte Sprache
  - einfacher Texteditor reicht theoretisch
- Für schnelle Hacks: ipython

```
→ ~ ipython
Python 3.9.16 (main, Dec 7 2022, 10:06:04)
Type 'copyright', 'credits' or 'license' for more information
IPython 8.2.0 -- An enhanced Interactive Python. Type '?' for help.

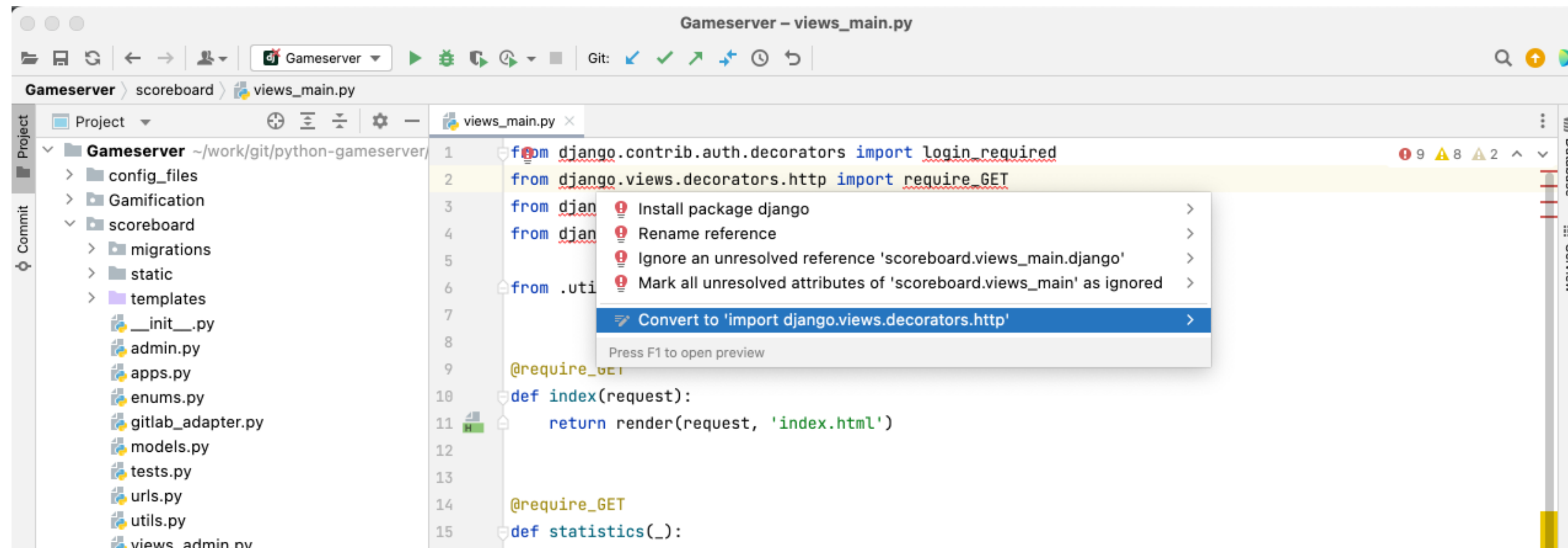
In [1]: print(2 ** 8)
256

In [2]:
```

- Schöner und persistent: Jupyter Notebook / Lab
  - web-basierte Oberfläche, analog zu ipython
  - speichert Ausgaben und lässt sich inkl. Ergebnissen teilen

# Python - wie code ich?

- Vollwertige IDE nützlich für größere Projekte
- Unsere Empfehlung: PyCharm (Community Edition kostenlos für alle)
  - Syntax-Highlighting, Auto-Completion, Auto-Bug-Fixing, ...





```

181         scale,
182         min=0.01, max=1000.0,
183         default=1.0,
184     )
185
186 def execute(self, context):
187
188     # get the folder
189     folder_path = (os.path.dirname(self.filepath))
190
191     # get objects selected in the viewport
192     viewport_selection = bpy.context.selected_objects
193
194     # get export objects
195     obj_export_list = viewport_selection
196     if self.use_selection_setting == False:
197         obj_export_list = [i for i in bpy.context.scene.objects]
198
199     # deselect all objects
200     bpy.ops.object.select_all(action='DESELECT')
201
202     for item in obj_export_list:
203         item.select = True
204         if item.type == 'MESH':
205             file_path = os.path.join(folder_path, "{}.obj".format(item.name))
206             bpy.ops.export_scene.obj(filepath=file_path, use_selection=True,
207                                     axis_forward=self.axis_forward_setting,
208                                     axis_up=self.axis_up_setting,
209                                     use_animation=self.use_animation_setting,
210                                     use_mesh_modifiers=self.use_mesh_modifiers_setting,
211                                     use_edges=self.use_edges_setting,
212                                     use_smooth_groups=self.use_smooth_groups_setting,
213                                     use_flags=self.use_flags_setting,

```

Let's code!

# Python Course GameServer

This Gameserver will assist you solving the exercises of the Python Course.

The [Scoreboard](#) shows how your performance relative to your fellow students.

If you are logged in you can see your details, as well as your current exam admission status, in the [User Panel](#).

To start with the exercises, please find them in the respective menu under [Exercises](#).

---

Howto: Gameserver