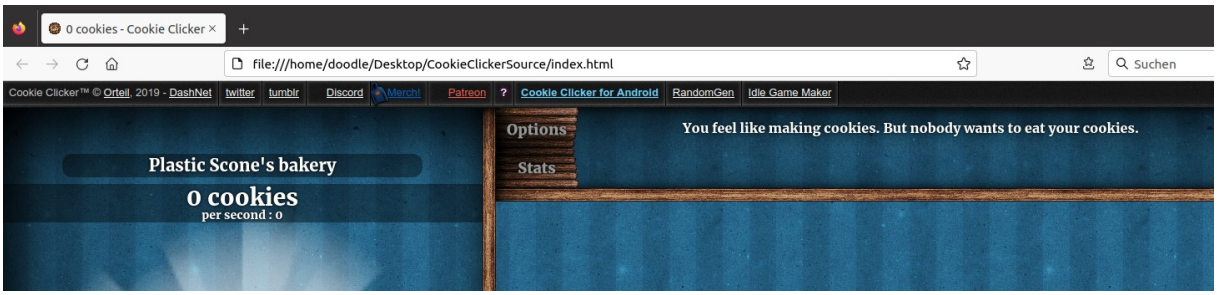
WPF-Python	AB: Die Cookieclicker-Mission	OSZ IMT
Name:	Klasse:	Datum:

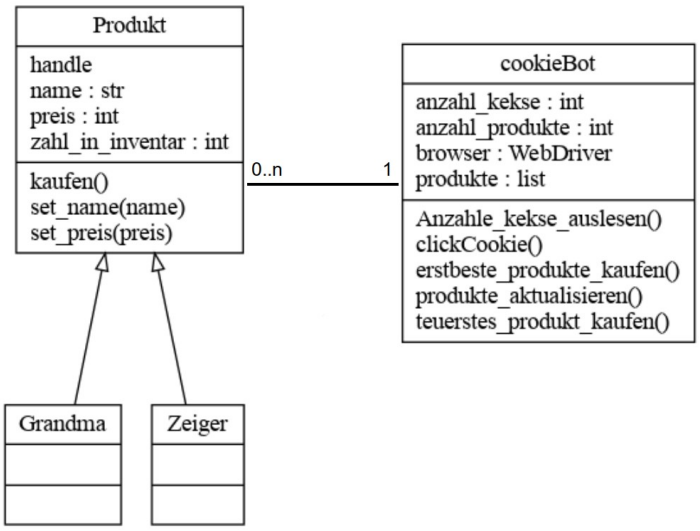
Erweiterung des Programms

Situation:

Glücklicherweise konnten Sie sich die Source-Datei des Cookie-Clicker-Spiels organisieren (Moodle). Diese lässt sich einfach im Browser öffnen und spielen. Klicken Sie auf die index.html Datei und öffnen Sie diese im Browser. Merken sie sich die url, die nun in Ihrem Browser angezeigt wird. Dies ist der lokale Pfad zu Ihrem Spiel.



Im Folgenden entwickeln Sie einen Bot nach folgendem Klassendiagramm. Der Bot kann Cookies clicken und die Spielprodukte kaufen.



Aufgaben:

Gegeben ist die Vorlage cookie_clicker_bot_Vorlage.py. Passen Sie die Vorlage entsprechend der Aufgaben an. Ziel ist es, dass der Bot, das Spiel für Sie übernimmt.

1. **Ermitteln** Sie alle notwendigen Bibliotheken, damit das Programm ausgeführt werden kann und fügen Sie diese am Kopf der Datei ein. [15 min.]
Lesen Sie hierzu **IB: Selenium Webdriver**

2. **Entwicklen** Sie die Methode clickCookie() für die Klasse „cookieBot“ [5 min.].

Hinweise:

- orientieren Sie sich an der Vorlage – particle_clicker.py und Verwenden Sie ggf. das Internet.
- nutzen Sie den Inspectionsmodus Ihres Browsers um die Cookie-Clicker-Html-Seite zu untersuchen und ermitteln Sie die ID des Cookies, um ihn finden zu können.
- Verwenden Sie die **find_element(By.ID,'bla') –** Methode des Webdrivers, um den Cookie zu finden
- nutzen Sie die click() Methode des elements um den Cookie zu klicken

3. Erstellen Sie eine Klasse Produkt entsprechend des Klassendiagramms. [10min.]

Hinweise:
- das Attribut handle erhält immer das Element (webdriver) zum clicken
- die Methode kaufen() führt dann die Methode self.handle.click() aus und erhöht die Inventarzahl

4. **Laden** Sie Ihr Ergebnis in die Abgabe hoch.

Zusatzaufgaben für Fortgeschrittene:

Schön und gut, die Cookies werden jetzt automatisch geklickt. Aber damit Ihr Chef sich endlich wieder vollständig auf die Firma konzentriert, wollen Sie auch die Einkäufe im Spiel automatisieren.

5. **Fügen** Sie die Methode Anzahl_kekse_aktualisieren() **hinzu**. [3min.]

```
def Anzahl_kekse_aktualisieren(self): #Guthaben
    soup = BeautifulSoup(self.browser.page_source, 'html.parser')
    titelelemente = soup.title.text.split(' ')
    if titelelemente[0].isdigit():
        self.anzahl_kekse = int(titelelemente[0])
    print(f"Anzahl an Cookies: {self.anzahl_kekse}")
```

Damit dies funktioniert muss die Python-Bibliothek „BeautifulSoup“ importiert werden.

Installation via Terminal: pip install beautifulsoup4
Import via: **from bs4 import BeautifulSoup**

6. **Erweitern** Sie Ihr Projekt so, dass Sie den cookie_clicker_bot als Python Modul verwenden und in eine Hauptdatei main.py importieren. [5min.]
Die Klassen aus dem gegebenen Klassendiagramm verbleiben in cookie_clicker_bot.py

7. **Entwickeln** Sie die Methode produkte_aktualisieren() [25 min.]

8. **Ergänzen** Sie Ihren Bot, um die Methode erstbeste_produkte_kaufen() [20 min.]