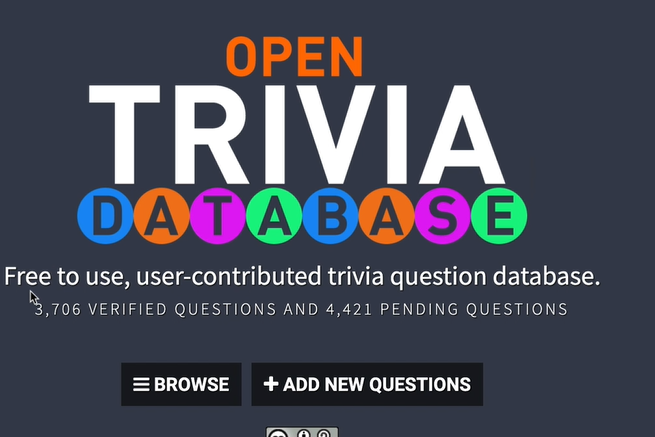
Day 34 – Quiz program and API practice



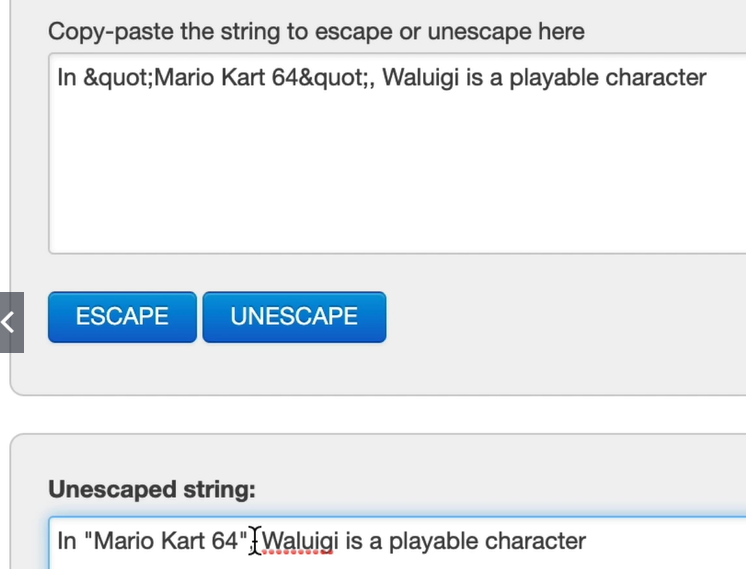
<https://opentdb.com/>

from question\_model import Question  
# from data import question\_data  
from quiz\_brain import QuizBrain  
import requests  
response = requests.get(url="https://opentdb.com/api.php?amount=10&difficulty=easy&type=boolean")  
response.raise\_for\_status()  
data=response.json()  
# print(data["results"])  
question\_data = data["results"]  
  
question\_bank = []  
for question in question\_data:  
 question\_text = question["question"]  
 question\_answer = question["correct\_answer"]  
 new\_question = Question(question\_text**,** question\_answer)  
 question\_bank.append(new\_question)  
  
  
quiz = QuizBrain(question\_bank)  
  
while quiz.still\_has\_questions():  
 quiz.next\_question()  
  
print("You've completed the quiz")  
print(f"Your final score was: {quiz.score}/{quiz.question\_number}")

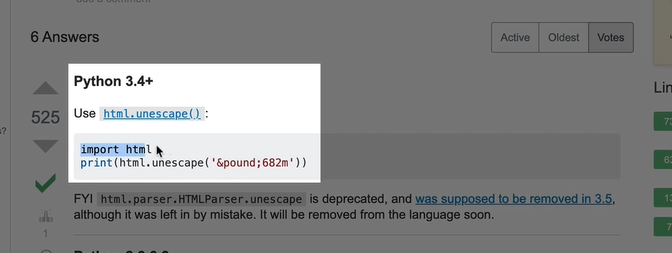
Correcting html entities

<https://www.w3schools.com/html/html_entities.asp>

<https://www.freeformatter.com/html-escape.html>

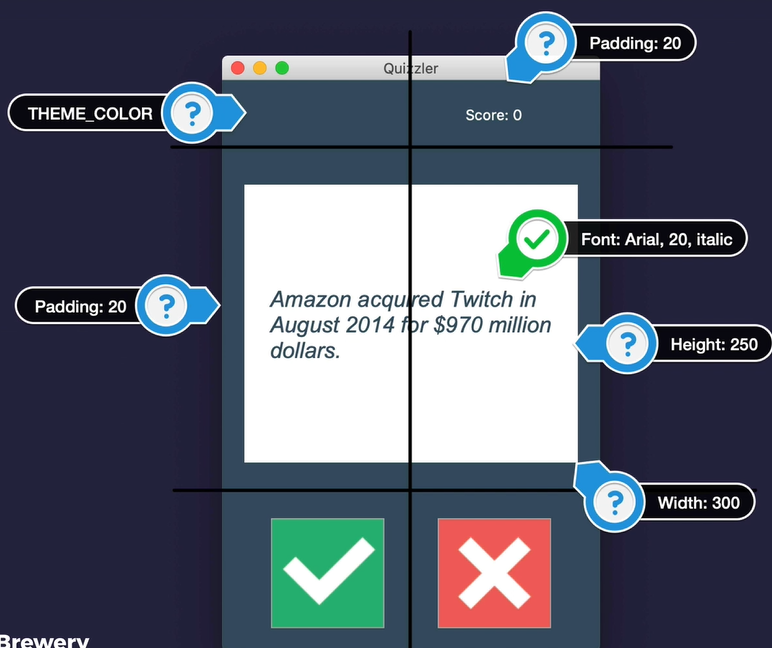


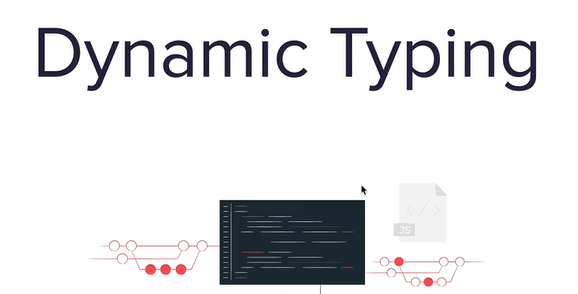
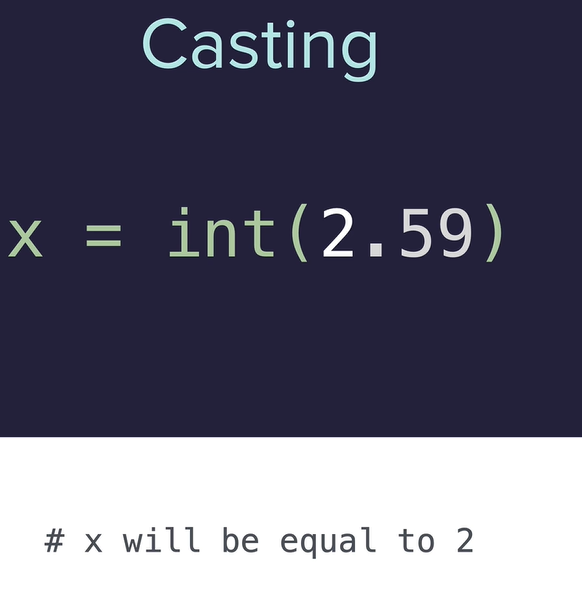
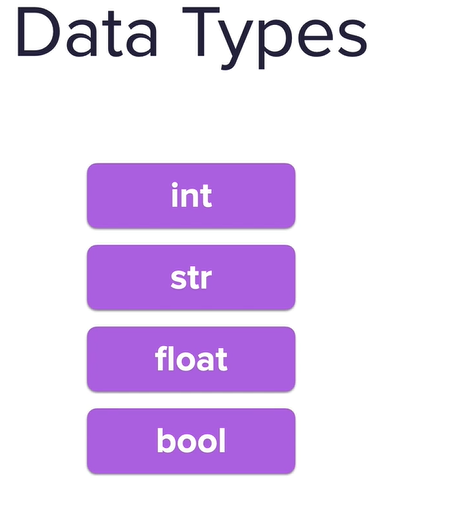
<https://stackoverflow.com/questions/2087370/decode-html-entities-in-python-string>



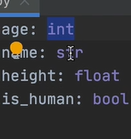
import html  
class QuizBrain:  
  
 def \_\_init\_\_(self**,** q\_list):  
 self.question\_number = **0** self.score = **0** self.question\_list = q\_list  
 self.current\_question = None  
  
 def still\_has\_questions(self):  
 return self.question\_number < len(self.question\_list)  
  
 def next\_question(self):  
 self.current\_question = self.question\_list[self.question\_number]  
 self.question\_number += **1** q\_text = html.unescape(self.current\_question.text)  
 user\_answer = input(f"Q.{self.question\_number}: {q\_text} (True/False): ")  
 self.check\_answer(user\_answer)

create user interface in a class using tkinter





Can declare them



And in a function 