کورس udemy – 100 days of code

Day001

میتونیم برای کمک گرفتن در کورس با #100daysofcode توی تویتر مطلب پیدا کنیم.



میخوایم the band name generator بسازیم.

Hello world رو رفت و داره استرینگ توضیح میده .

توی pycharm داره پیش میره ، و جالبیش اینه که توی Pycharm میتونیم اون هایلایتی که زیر چیزای اشتباه میندازه رو هاور کنیم و نکته ش رو ببینیم. فکر کنم این توی vscode نیست.

-نکته جدید: خط آخر توی اکثر زبان ها بهتره که خالی باشه. وگرنه ai که چک میکنه اشکال تشخیص میده و Char آخر رو Highlight میکنه .

-یادآوری: \n برای رفتن به خط بعدی در خروجی

Concat با +

یه نرم افزار معرفی کرد به نام thonny که میتونیم باهاش کدی که نوشتیم رو ببینیم مرحله به مرحله توسط کامپیوتر چطور اجرا میشه. این جالب میتونه باشه و در ادامه استفاده های زیادی ازش بکنیم . شاید Pycharm و vscode هم داشته باشنشون .

تا 012 Variables

ادامه

همونطور که میتونیم باید از underline استفاده کنیم به جای camelCase .

یادآوری: len() برای گرفتن تعداد char ها.

-توی vscode توی اجرای کد هایی که توی هرجایی بجز c ذخیره شده به مشکل خوردیم . توی همون pycharm پیش میریم. یا عشقمون کشید با jupyter میریم .

تا انتهای day001

ادامه

Day002

Len() برای Int عمل نمیکنه و میگه که no length . باید conversion انجام بدیم روی Int ها وعدد ها اگر میخوام Len بگیریم .

با گذاشتن براکت جلوی هر چیزی که Length داره میتونه خروجی Index مربوطه رو داشته باشیم

Index منفی هم میتونیم استفاده کنیم و از آخر میشماره

Concatenation

نکته جالب: میتونیم بین اعداد Underline بزاریم به جای کاما تا خوندشون راحت تر بشه

Type() برای چک کردن data type

Type conversion در یسری حالتا غیر قابل انجامه.



یادآوری: اولی Float میده به صورت پیشفرض ، دومی ceil میکنه

\*\* توان

Round() روی اعداد >> گرد میکنه بسته به مقدار اعشار به بالا و یا به پایین 5 بشه به بالا

Round(x,x) مقدار دوم میتونیم اعلام کنیم تا چند رقم اعشار گرد کن.

Fstring : print(f”string{value}”

تمرین امروز



تا وسطای 009 Day 2 Project Tip Calculator

ادامه

ما برای تمرین روز دوم از number: .2f استفاده کردیم تا اعشارش رو محدود کنیم ولی خودش از round(x, x) استفاده کرده. اولی متغیری هست که میخوایم روند شه ، دومی مقدار روند کردنشه.

Day003 treasue-islad-end

تا 003 Control Flow with if else and Conditional Operators

ادامه از 003 Control Flow with if else and Conditional Operators

Modulo operator یا همون % خودمون. باقیمانده.

تمرین:



Nested if/else

تودرتو









-داخل pycharm یه سیستم تشخیص simplify داریم که میتونیم روش کلیک کنیم و همون کدی که نوشتیم رو ساده ترش رو بهمون بده .



<https://ascii.co.uk/art>

یادآوری: برای اینکه بتونیم یچیزی مثل صندوق ابتدای تمرین نهایی رو print() کنیم باید به جای “” از ‘’’ ‘’’ استفاده کنیم که 3 تا Single quotation میشه. این اجازه میده چنیدن خط رو یکجا پرینت بگیریم.

پروژه نهایی هم در انتها اینه که یه بازی ابتدا بر اساس diagram زیر بسازیم. بعدش اگر حست کشید بزرگتر و بسط داده ترش کن یا یکی دیگه بساز با یه diagram دیگه .



پایان day002

(پروژه day003 بازی سنگ کاغذ قیچیه. که بیشتر با مفهوم randomization و Python Lists قراره کار کنیم.)

ادامه

Day004

پروژه نهایی ساخت یک بازی سنگ کاغذ قیچی

Randomization

برای اینکه بتونیم از مقادیر random استفاده کنیم باید از module اضافه کنیم به برنامه :

Import random

که از.random() و .randint() استفاده میکنیم برای گرفتن یک مقدار random که البته فرق کوچیکی دارن که با یه سرچ در میاد (مثلا random خودِ صفر رو در نظر میگیره تا اون عددی که بهش دادی)

مفهوم module : میتونیم بخش هایی از برنامه رو



میتونیم ضرب هم کنیم که دو parameter رو ضرب میکنه از اونجایی که یکیشون صفره اونیکی فقط اعمال میشه عملا

یه برنامه نوشتیم واسه "شیر یا خط" که یه سکه پرت میکنیم.

Python list

یه data structure عه .



یادآوری: با Index میتونیم یک item رو return کنیم. با مقدار منفی از انتها شمرده میشه.

aList.add() و aList.extend() و...

میتونیم با random.choice() از item های یک List به صورت random returnداشته باشیم.

Nested list : میتونیم داخل یک List ، List تعریف کنیم (داخل index ها)

تا 007 day 004 03:17

ادامه

برای اینکه بتونیم برنامه ای که نوشتیم رو کاری کنیم دائم در حال اجرا باشه و در صورت اینکه کاربر قصد پایان داشت خاتمه پیدا کنه باید از ساختار کلی زیر در پایتون استفاده کنیم (که روی پروژه روز 4 سنگ کاغذ قیچی پیاده ش کردیم) :



حالا باید بریم ببینیم روش خودش چیه

1-ascii art هارو داخل یک List ذخیره کرد که با مقادیر وارد شده توسط user مطابقت داره که باعث میشه فقط بنویسه List\_name [user\_choice] که برای پرینت کردن شکل ها خیلی راحت تره. (کاری که ما کردیم این بود که یه module براش ساختیم و هر دفعه برای پرینت کردن هرکدوم باید این رو مینوشتیم: module\_name.scissors و یدور هم نیاز به Import module\_name داشت. که خب کاری که خودش کرده راحت تره)

ادامه

جلسه قبل در طی ور رفتن با گیت فایل هایی که نوشته بودیم پرید ، ولی از دوباره یه repository ساختیم که اونجا Push کنیم و نیاز نیست از دوباره کد هارو بزنیم مربوط به ابتدا رو ، فقط از اینجا به بعد رو انجام میدیم .

Day005

پروژه امروز PyPassword generator هست .



جالب: یه سایت معرفی کرد برای اینکه چک کنیم email رو و اگر شرکتی مورد هک واقع شده و اطلاعات شما هم اونجا بوده و رفته بهتون بگه. ([link](https://haveibeenpwned.com/))

خب ما دوتا راه داریم این کارو انجام بدیم: easy version و hard version

ابتدا میریم سراغ easy .

در easy به ترتیبِ حروف ، سمبل، عدد رفتیم ولی در hard باید بدون ترتیب مشخص پسوورد ساخته میشد. که ما از خودمون با random درستش کردیم. ولی داخلِ خودِ random یه method هست به نام shuffle() که همین کار رو انجام میده .