

Year 11 Homework Python Challenges

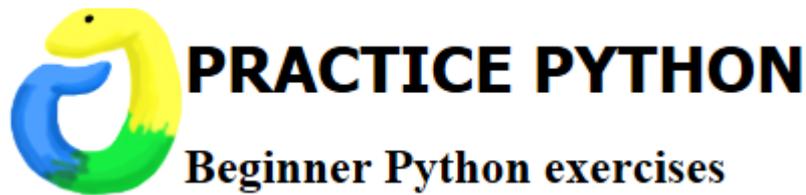


NEWMAN R.C. COMPUTING DEPARTMENT



NAME:

Beginner Python Challenges



Go to the website:-

<https://www.practicepython.org/>

Work your way through the python challenges. Screenshot and annotate your solutions.

- 1: [Character Input](#) 🐍
- 2: [Odd Or Even](#) 🐍
- 3: [List Less Than Ten](#) 🐍🐍
- 4: [Divisors](#) 🐍🐍
- 5: [List Overlap](#) 🐍🐍
- 6: [String Lists](#) 🐍🐍
- 7: [List Comprehensions](#) 🐍🐍
- 8: [Rock Paper Scissors](#) 🐍🐍🐍
- 9: [Guessing Game One](#) 🐍🐍🐍
- 10: [List Overlap Comprehensions](#) 🐍🐍
- 11: [Check Primality Functions](#) 🐍🐍🐍
- 12: [List Ends](#) 🐍
- 13: [Fibonacci](#) 🐍🐍
- 14: [List Remove Duplicates](#) 🐍🐍
- 15: [Reverse Word Order](#) 🐍🐍🐍
- 16: [Password Generator](#) 🐍🐍🐍🐍
- 17: [Decode A Web Page](#) 🐍🐍🐍🐍
- 18: [Cows And Bulls](#) 🐍🐍🐍
- 19: [Decode A Web Page Two](#) 🐍🐍🐍🐍
- 20: [Element Search](#) 🐍
- 21: [Write To A File](#) 🐍
- 22: [Read From File](#) 🐍
- 23: [File Overlap](#) 🐍🐍

- 24: [Draw A Game Board](#) 🐍🐍
- 25: [Guessing Game Two](#) 🐍🐍🐍
- 26: [Check Tic Tac Toe](#) 🐍🐍
- 27: [Tic Tac Toe Draw](#) 🐍🐍
- 28: [Max Of Three](#) 🐍
- 29: [Tic Tac Toe Game](#) 🐍🐍🐍
- 30: [Pick Word](#) 🐍🐍
- 31: [Guess Letters](#) 🐍🐍
- 32: [Hangman](#) 🐍🐍
- 33: [Birthday Dictionaries](#) 🐍
- 34: [Birthday Json](#) 🐍🐍
- 35: [Birthday Months](#) 🐍🐍
- 36: [Birthday Plots](#) 🐍🐍🐍

Challenge

Date Completed

Signed

- 1: [Character Input](#) 🐾
- 2: [Odd Or Even](#) 🐾
- 3: [List Less Than Ten](#) 🐾🐾
- 4: [Divisors](#) 🐾🐾
- 5: [List Overlap](#) 🐾🐾
- 6: [String Lists](#) 🐾🐾
- 7: [List Comprehensions](#) 🐾🐾
- 8: [Rock Paper Scissors](#) 🐾🐾🐾
- 9: [Guessing Game One](#) 🐾🐾🐾
- 10: [List Overlap Comprehensions](#) 🐾🐾
- 11: [Check Primality Functions](#) 🐾🐾🐾
- 12: [List Ends](#) 🐾
- 13: [Fibonacci](#) 🐾🐾
- 14: [List Remove Duplicates](#) 🐾🐾
- 15: [Reverse Word Order](#) 🐾🐾🐾
- 16: [Password Generator](#) 🐾🐾🐾🐾
- 17: [Decode A Web Page](#) 🐾🐾🐾🐾
- 18: [Cows And Bulls](#) 🐾🐾🐾
- 19: [Decode A Web Page Two](#) 🐾🐾🐾🐾
- 20: [Element Search](#) 🐾
- 21: [Write To A File](#) 🐾
- 22: [Read From File](#) 🐾
- 23: [File Overlap](#) 🐾🐾
- 24: [Draw A Game Board](#) 🐾🐾
- 25: [Guessing Game Two](#) 🐾🐾🐾
- 26: [Check Tic Tac Toe](#) 🐾🐾
- 27: [Tic Tac Toe Draw](#) 🐾🐾
- 28: [Max Of Three](#) 🐾
- 29: [Tic Tac Toe Game](#) 🐾🐾🐾
- 30: [Pick Word](#) 🐾🐾
- 31: [Guess Letters](#) 🐾🐾
- 32: [Hangman](#) 🐾🐾
- 33: [Birthday Dictionaries](#) 🐾
- 34: [Birthday Json](#) 🐾🐾

More Advanced Python Challenges

Mastermind

Generate a random four digit number. The player has to keep inputting four digit numbers until they guess the randomly generated number. After each unsuccessful try it should say how many numbers they got correct, but not which position they got right. At the end of the game should congratulate the user and say how many tries it took.

Extensions:

1. Let the user pick an easy mode which shows the user which position that they guessed correctly
2. Let the user pick a hard mode that gives five digits instead of four
3. After the game is finished, ask the user for their name and input their score into a table. Show them the high score at the start of the game so that it gives a sense of competition.

Averages

Make a program that asks the user for a series of numbers until they either want to output the average or quit the program.

Extensions:

1. Expand the program to print the median and mode averages also
2. Include options so that if the user wants to, they can save their list of numbers to a text file and read them back out later on.

Email validator

Make a program to check whether an email address is valid or not.

For instance, you could make sure that there are no spaces, that there is an @ symbol and a dot somewhere after it. Also check the length of the parts at the start, and that the end parts of the address are not blank.

Extensions:

1. When an email address is found to be invalid, tell the user exactly what they did wrong with their email address rather than just saying it is invalid
2. Allow the user to choose to give a text file with a list of email addresses and have it process them all automatically.

Password reset program

Only accept a new password if it is:

1. At least eight characters long
2. Has lower case and upper case letters.

The password reset program should also make the user input their new password twice so that the computer knows that the user has not made any mistakes when typing their new password.

Extensions:

1. Make some sort of algorithm to suggest how strong the password is (Weak, Medium, Strong) depending on length, whether or not the password has special characters in etc
2. Let the user input their username. The program should go to a text file with a list of usernames and old passwords, and the program should only let you change your password if you input your old password.

Basic lists

Make a program that lets a user input a series of names into a list. The program should then ask the user whether they want to print out the list in the original order, or in reverse.

Extensions:

1. Enable the user to choose what number item in the list they want to print out
2. Enable the user to only print out a 'slice' of the list (eg item three to item nine only)
3. Enable the user to remove any items of the list that they want to
4. Enable the user to save their list to a file for later, and also enable them to load it back up again too
5. 'Clean' the list by making all the items lowercase.

Max and min list

Write a program that lets the user input a list of numbers. Every time they input a new number, the program should give a message about what the maximum and minimum numbers in the list are.

Extensions

1. The program should let the user choose the allowable minimum and maximum values, and should not let the user input something to the list that is outside of these bounds
2. The user should be able to write these values to a file and then also read them back out again.
3. If a file has any numbers outside of the boundaries, it should strip them out of the list once it has read them in.

Letter list

Write a program that lets a user choose a letter. The program will then find all the words beginning with that letter in a list and print them out. It should also say how many words it found.

Extensions:

1. Let the user load up a list of words from a file and have the program process them all
2. Change the program so that the user can choose whether they want all words with only the start of the letter, or ANY place in the word.

RPG character/Pokemon stat creator

Make a program which will randomly create a character's stats based on several rules set by the user. Have it generate a class, gender, strength/magic/dexterity points, and extra abilities or trades. Have the program save it to a file which can then be printed out so that it can be used in a game.

Extension:

1. Make a mystical name generator. Perhaps randomise different name parts such as sha-ra-lam or big-lim-con to create names for each of your randomly generated characters.

Quiz Maker

Make an application which takes various questions from a file, picked randomly, and puts together a quiz for students, and then reads a key to grade the quizzes. Each quiz can be different.

0 Check if Palindrome

Checks if the string entered by the user is a palindrome. A palindrome is a word that reads the same forwards as it does backwards like "racecar".

1	Count Words in a String Counts the number of individual words in a string. For added complexity, the program could read these strings in from a text file and generate a summary.
2	Pig Latin Pig Latin is a game of alterations played on the English language game. To create the Pig Latin form of an English word the initial consonant sound is transposed to the end of the word and an ay is affixed (Ex.: "banana" would yield anana-bay). Read Wikipedia for more information on rules (http://en.wikipedia.org/wiki/Pig_Latin).
3	Count Vowels Enter a string and the program counts the number of vowels in the text. For added complexity have it report a sum of each vowel found.
4	Unit Converter (temperature, currency, volume, mass and more) Converts various units between one another. The user enters the type of unit being entered, the type of unit they want to convert to and then the value. The program will then make the conversion.
5	Change Return Program The user enters a cost and then the amount of money given. You should write a program that works out what denominations of change should be given in pounds, 50p, 20p, 10p etc. Extensions: <ol style="list-style-type: none">1. The program will figure out the change for the American currency and the number of quarters, dimes, nickels, pennies needed for the change2. Make an automatic testing part of your program where it automatically generates a random price and an amount that you give the cashier. It then works out what change to give, and then tests that your program works by adding the change back onto the price of the item to prove your program works. It should flag an error if there are problems.

6	<p>Shopping list</p> <p>Create a program that will keep track of items for a shopping list. The program should allow you to keep adding new items. You should also be able to record which shop you are going to visit for the item.</p> <p>Extensions:</p> <ol style="list-style-type: none">1. Extend the program to record what priority the item is2. Extend the program to record whether or not you have bought the item or not3. Extend the program to say how much you are willing to pay for each item4. Extend the program to say the quantity needed for each item5. Make a part of your program that can give you the approximate total of what you intend to spend on your shopping trip. Make sure that you take into account whether you have already bought the item or not (ie don't add this up!).
7	<p>Hangman</p> <p>Create a version of the Hangman game using Lists. The program should ask for the word to guess and the number of chances to be given.</p> <p>For every guess you should update the user with which letters they have guessed incorrectly, as well as replacing the letters in the guess word with the ones they have guessed correctly.</p> <p>You should also show the user how many chances they have left.</p> <p>Extension:</p> <ol style="list-style-type: none">1. Make sure that you do not let the user lose a life if they guess a letter that they have already guessed.
8	<p>Squares</p> <p>Create a program that will ask the user for a number and then print out a list of numbers from 1 to the number entered and the square of the number. For example, if the user entered '3' then the program would output:</p> <p>1 squared is 1 2 squared is 4 3 squared is 9</p>

9

Times tables

Create a program which will produce the times table for a number entered by the user

eg if the user enters '2' it should produce:

$$1 \times 2 = 2$$

$$2 \times 2 = 4$$

$$3 \times 2 = 6$$

0

Binary/Hexidecimal/Decimal

Create a program which will convert a given decimal up to 255 into its 8-bit binary equivalent.

Extensions:

1. Extend program 7 to convert the binary number to hexadecimal
2. Create a program to convert from hexadecimal to decimal.

MARKING GRID

<u>ACTIVITY</u>	<u>GRADE (1-9)</u>	<u>EVIDENCE?</u>	<u>COMMENTS</u>
MASTERMIND			
AVERAGES			
EMAIL VALIDATOR			
PASSWORD RESET PROGRAM			
BASIC LISTS			
MAX AND MIN LIST			
LETTER LIST			
RPG CHARACTER/POKEMON STAT CREATOR			
QUIZ MAKER			
CHECK IF PALINDROME			
COUNT WORDS IN A STRING			
PIG LATIN			
COUNT VOWELS			
UNIT CONVERTER (TEMP, CURRENCY, VOLUME, MASS AND MORE)			
CHANGE RETURN PROGRAM			
SHOPPING LIST			
HANGMAN			
SQUARES			
TIMES TABLES			
BINARY/HEXIDECIMAL/DECIMAL			

OVERALL GRADE:

WHAT WENT WELL?

**WHAT DO I NEED TO
IMPROVE ON?**