Cheatography

data3500 Cheat Sheet

by chase2981 via cheatography.com/135397/cs/40897/

List numbers one by one

```
# Get input from the user
number = int(input("Please enter a number 7 to 10 digits: "))

# Determine the number of digits in the integer
num_digits = len(str(number))

# Ensure the number is between 7 to 10 digits
if 7 <= num_digits and num_digits <= 10:
    # Use a loop to pick off each digit
    for i in range(num_digits):
        # print('num_digits', num_digits, '10 ** num_digits', 10 ** num_digits)

        # Calculate the power of 10 for the current position
        power_of_10 = 10 ** (num_digits - 1)

        # print('power_of_10', power_of_10, 'number', number, 'number // power_of_10', number // power_of_10)

        # Extract the leftmost digit
        digit = number // power_of_10

        # Print the digit
        print(digit)

        # Remove the leftmost digit from the number
        number = number % power_of_10

        # Decrease the number of digits left to process
        num_digits -= 1
else:
    print("Number was not 7 to 10 digits.")
```

sentinel loop

```
# Initialize total miles and total gallons to zero
total_miles = 0
total_gallons = 0
overall_mpg = 0

# Start the sentinel-controlled loop, meaning it repeats until it is broken by user input
while True:
    # Get the gallons used from the user, convert to float
    gallons = float(input("Please enter the gallons used (-1 to stop the loop): "))

    # Check for the sentinel value and if so escape
    if gallons == -1:
        break

    # Get the miles driven from the user
    miles = float(input("Please enter the miles driven: "))

    # Calculate miles per gallon for this tankful
    mpg = miles / gallons
    print("The miles/gallon for this tank was {0}".format(mpg))

    # Update the total miles and total gallons for avg calculation later
    total_miles += miles
    total_gallons += gallons

# Calculate the overall average miles per gallon
if total_gallons != 0:
    overall_mpg = total_miles / total_gallons

print("The overall average miles/gallon was {0}".format(overall_mpg))
```

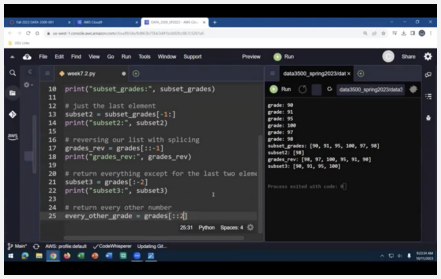
palindrome

```
# 3.12 (Palindromes) A palindrome is a number, word or text phrase that reads the same

# Get input from the user
number = int(input("Please enter a five-digit integer: "))

# Ensure the number is five digits
if 10000 <= number and number <= 99999:

    # Extract the digits
    digit1 = number // 10000
    print('digit1', number // 10000)
    digit2 = (number % 10000) // 1000
    print('digit2', number % 10000, 'divide by 1000', (number % 10000) // 1000)
    digit3 = (number % 1000) // 100
    print('digit3', number % 1000, 'divide by 100', (number % 1000) // 100)
    digit4 = (number % 100) // 10
    print('digit4', number % 100, 'divide by 10', (number % 100) // 10)
    digit5 = number % 10
    print('digit5', number % 10)

    # Check if the number is a palindrome
    if digit1 == digit5 and digit2 == digit4:
        print(f"{number} is a palindrome.")
    else:
        print(f"{number} is not a palindrome.")
else:
    print("Please enter a five-digit integer.")
```

consecutives in list

```
i = 1
total = 0
consecutive_3_14 = 0
consecutive_3_141 = 0
iteration_for_3_14 = 0
iteration_for_3_141 = 0

for denominator in range(1, 6000, 2):
    if i % 2 == 0:
        total -= 4/denominator
    else:
        total += 4/denominator

    # Check for consecutive 3.14 approximations
    if str(total)[:4] == '3.14':
        consecutive_3_14 += 1
        if consecutive_3_14 == 2 and not iteration_for_3_14:
            iteration_for_3_14 = i
            print('hit 3.14', i)
    else:
        consecutive_3_14 = 0

    # Check for consecutive 3.141 approximations
    if str(total)[:5] == '3.141':
        consecutive_3_141 += 1
        if consecutive_3_141 == 2 and not iteration_for_3_141:
            iteration_for_3_141 = i
            print('hit 3.141', i)
    else:
        consecutive_3_141 = 0

    # print('i:', i, 'total', total)
    i += 1
```

splicing



moving avg

arithmetic

pi

odd or even

multiples

formatting

By **chase2981**

cheatography.com/chase2981/

Not published yet.

Last updated 19th October, 2023.