# Урок 12.01. Методы insert, remove, index

С каждым разом мы работаем с всё более сложными типами и структурами данных и тем самым изучаем всё больше проблем, с которыми сталкивается разработчик. Однако, как мы уже поняли, многие проблемы решаются просто. Так как об их решении уже кто-то позаботился. И давайте рассмотрим такую задачу.

Задача «Топ языков»

По информации из интернета и по собственному мнению мы составили список языков программирования, и туда входят: Python, Java, JavaScript, SQL. И внезапно мы обнаружили, что забыли включить в этот список язык С++, который должен стоять после Java. Нам нужно исправить это недоразумение. Пока что мы будем считать, что знаем, в каком месту стоит Java.

langs = ['Python', 'Java', 'JavaScript', 'SQL']

# И нам нужно вписать C++ между Java и JavaScript

# По условию задачи нам известно куда нужно вписать, между вторым и третьим местом

# Давайте создадим для этого новый список и туда вставим

new\_langs = []

for i\_lang in range(2):

 new\_langs.append(langs[i\_lang])

new\_langs.append('C++')

for i\_lang in range(2, len(langs)):

 new\_langs.append(langs[i\_lang])

print(new\_langs)

Всё конечно работает, но наш код опять очень громоздкий и решает простую и довольно таки распространенную задачу. И как вы уже догадались, программисты давно решили и такую задачу. И всё ээто можно заменить одной строчкой

langs = ['Python', 'Java', 'JavaScript', 'SQL']

langs.insert(2, 'C++')

print(langs)

И тут команда insert реализует вставку объекта на то место индекс, которого мы указали. Т.е. мы указываем индекс куда надо поставить новый объект, python вставляет его туда, а все элементы которые шли дальше сдвигает «вправо» на один элемент. Тут нужно сделать важно уточнение, команды append и insert правильно называть методами. И если после имени списка ввести точку, то мы с вами увидим еще много методов, которые доступны нам. Более того у разных типов данных имеются свои методы, некоторые из которых могут пересекаться. Давайте изучим еще несколько методов для списков.

И дополним нашу задачу следующим условием: пусть пользователь вводит новый язык программирования и вводит место куда его нужно вставить.

langs = ['Python', 'Java', 'JavaScript', 'SQL']

print(langs)

user\_lang = input('Введите новый язык программирования: ')

user\_after\_lang = input('После какого языка вставить: ')

# И тут мы должны написать цикл по поиску индекса языка программирования,

# после которого хотим вставить новый

# но зачем если эта задача уже тоже решена

i\_lang = langs.index(user\_after\_lang)

langs.insert(i\_lang + 1, user\_lang)

print(langs)

Введем C++ и вставим после Java. Хорошо, всё получилось. А что будет, если ввести несуществующий элемент, например, PHP… А тут python выдал нам ошибку, потому что такого элемента в списке нет.

Теперь давайте возьмем одну из задач домашнего задания «Рейтинг фильма» и немного изменим. Мы поддерживаем свой собственный киносайт, и хотим, чтобы после регистрации пользователи могли составить свой собственный рейтинг фильмов из тех, которые есть на сайте. Т.е. нужно реализовать программу, которая позволяет добавлять свой рейтинг к фильму, удалять их оттуда, или вставлять на нужное место. Сделаем такую программу.

def is\_film\_exist(movie, films\_list):

 for i\_movie in films\_list:

 if movie == i\_movie:

 return True

 return False

films = [ 'Назад в будущее', 'Дюна', 'Дюна 2', 'Человек-паук', 'Мстители', 'Матрица', 'Начало', 'Отступники', 'Джентельмены', 'Игра Эндера', ]

my\_list = []

while True:

 print(f'\nВаш текущий топ фильмов: {my\_list}')

 new\_movie = input('Введите название фильма: ')

 if is\_film\_exist(new\_movie, films):

 print('Команды: добавить, удалить, вставить')

 command = input('Введите команду: ')

 if command.lower() == 'добавить':

 my\_list.append(new\_movie)

 elif command.lower() == 'удалить':

 if is\_film\_exist(new\_movie, my\_list):

 my\_list.remove(new\_movie)

 else:

 print('Такого фильма нет в нашем рейтинге')

 elif command.lower() == 'вставить':

 ind = int(input('На какое место вставить: '))

 my\_list.insert(ind - 1, new\_movie)

 else:

 print('Такого фильма нет на сайте.')

Протестируем нашу программу. Всё работает, только наша программа чувствительна в регистру. Но ничего страшного вы сами уже можете исправить это.

Итого мы с вами познакомились с новыми методами списков: insert, remove, index.