**Задание 1**

Напишите программу, которая запрашивает ввод двух значений. Если хотя бы одно из них не является числом, то должна выполняться конкатенация, т. е. соединение, строк. В остальных случаях введенные числа суммируются.

|  |  |
| --- | --- |
| **Ввод:** | **Ввод:** |
| Первое значение: 4  Второе значение: 5 | Первое значение: а  Второе значение: 5 |
| **Вывод:** | **Вывод:** |
| Результат: 9.0 | Результат: а5 |

num=input("Первое значение ")  
num2=input("Второе значение ")  
try:  
 res=float(num)+float(num2)  
 print(res)  
except:  
 print(num+num2)

**Задание 2**

Напишите программу для проверки валидности номера телефона.

Главный критерий: номер может содержать в себе только целые числа.

Программа запрашивает ввод через input().

|  |  |
| --- | --- |
| **Ввод:** | **Ввод:** |
| Insert your phone number: 87777777777 | Insert your phone number: 123412qw2e |
| **Вывод:** | **Вывод:** |
| We will contact you by phone number: 8777777777 | Phone number can contain only digits! |

num=input("Insert your phone number: ")  
try:  
 res=int(num)  
 print("We will contact you by phone number ", res)  
  
except:  
 print("Phone number can contain only digits! ")

**Задание 3**

Напишите программу которая запрашивает числа пока мы не введем 0. Все введенные числа кроме 0 должны записываться в массив.

После программа запрашивает ввести целое число как индекс. Если в списке есть элемент с введенным индексом, программа выведет данный элемент,  если нет, выведет “There is no such element”.

|  |  |
| --- | --- |
| **Ввод:** | **Ввод:** |
| 2 3 4 5 6 7 8 0  Insert index: 3 | 1 2 3 4 5 6 7 8 9 0  Insert index: 11 |
| **Вывод:** | **Вывод:** |
| 5 | There is no such element |

c=[]  
arr = list(map(int,input().split()))  
b = int(input("Insert index:"))  
for i in range(len(arr)):  
 if arr[i]!=0:  
 c.append(arr[i])  
try:  
 print(c[b])  
except:  
 print("There is no such element")

**Задание 4**

На этой задаче мы будем отлавливать ошибки для пустых объектов.

Создайте класс:

class **Book**:

    int number

    String title

    String author

    int pages

Создайте конструктор и функцию для вывода информации.

Далее в программе нужно создать 5 объектов книг и один объект укажите пустым, то есть:

book = None

(В Python пустота обозначается как None)

Все объекты, включая пустую книгу отправляем в список, и с помощью цикла выводим все данные. Для пустого объекта выводим сообщение “Book if empty”.

class Book:  
 def \_\_init\_\_(self, number, title, author, pages):  
 self.number=number  
 self.title=title  
 self.author=author  
 self.pages=pages  
  
 def getData(self):  
 return self.number, self.title, self.author, self.pages  
  
b1 = Book(1, "Игра престолов", " Р.Р.Мартин", 500)  
b2 = Book(2, "Чудо на гудзоне", "Петров П.", 350)  
b3 = Book(3, "Билли Милиган", "Петров П.", 600)  
b4 = Book(4, "Несломленнные", "Петров П.", 900)  
b5 =None  
b = [b1, b2, b3, b4, b5]  
  
for i in b:  
 try:  
 print(i.getData())  
 except:  
 print("Book if empty")

Отправить на проверку

### Задание 5

Создайте мини программу, в которой я ввожу свои данные через консоль, и заполняю массив пользователей.

У меня имеется класс пользователей **User**

               str name

               str surname

           int age

А также конструктор для всех параметров.

В главном классе, то есть основной части,  я имею список пользователей, с максимальным размером 5 человек. Моя программа запрашивает пользователя ввести этих 5 пользователей через консоль.

Однако,  если я во время введения возраста ввожу неверные данные (не целостное число), то программа должна автоматом ввести число 0 в поле возраста и добавить объект в список.

В конце ввода  программа должна вывести среднее значение возраста в списке.

class User:  
 def \_\_init\_\_(self, name, surname, age):  
 self.name=name  
 self.surname=surname  
 self.age=age  
  
 def getData(self):  
 return self.name, self.surname, self.age  
u=[]  
avg=0  
  
for i in range(0,5):  
 name=input("Введите имя ")  
 surname=input("Введите фамилию ")  
 age=input("Введите возраст ")  
  
 try:  
 age=int(age)  
 except:  
 age=0  
  
 avg+=age  
 u.append(User(name, surname, age))  
  
avg=avg/len(u)  
  
for i in u:  
 print(i.getData())  
print(avg)