# Задание: Операции с строками

**Инструкции к заданию**

Получить python код, который делает следующее:

1. дано строку " c34o56o78l12!12!12!", нужно получить копию состоящую из каждой третьей буквы оригинальной строки
2. дано строку "this is my home", нужно получить копию строки, где home нужно заменить на family
3. дано строку "your balance is 20.0", нужно получить копию строки, где будет только 4 последних символа
4. дано строку "codexxcode", нужно посчитать сколько раз повторяется подстрока "code"

**Ожидаемый результат**

Python код для каждого пункта задания

**Подсказки**

Ключевые концепции и на что стоит обращать внимание:   
string, slicing, string.find(), string.replace(), string.count()

**Правильный ответ**

1. " c34o56o78l12!12!12!"[::3]
2. "this is my home".replace("home", "family")
3. "your balance is 20.0"[-4:]
4. "codexxcode".count("code")

# Задание: Операции с числами

**Инструкции к заданию**

Получить python код, который делает следующее:

* получить остаток от деления 10 на 3
* от сложения 100 и 133, вычислить 20% и возвести результат в 3ю степень
* сгенерировать случайное вещественное число и округлить его до второго знака после точки

**Ожидаемый результат**

Python код для каждого пункта задания

**Подсказки**

Ключевые концепции и на что стоит обращать внимание:   
numbers, модуль math, модуль random, divmod(), %, /, //

**Правильный ответ**

1. 10 % 3
2. ((100 + 133) \* 0.2) \*\* 3
4. import random
5. round(random.uniform(100, 200), 2)

# Задание: Операции с последовательностями

**Инструкции к заданию**

Cоздать массив (и заполнить его случайными целыми числами от 0 до 10, если чувствуете свои силы в python, если нет, то просто пропустите), например [[1,2,3],[9,7,5],[4,6,8]], далее написать код:

* создания массива
* заполнение массива случайными числами
* вычисление суммы двух последних значений первого вложенного списка (5)
* получить минимальное значение по второй колонке (2,7,6)
* получить максимальное значение всей матрицы (9)
* узнать порядковый номер вложенного списка, в котором встречается число 4 (2)
* вывести данные в формате "last two elements is:" "min of second row is " "max of whole array is " "index of sub-array where 4 occured is"

**Ожидаемый результат**

Рython код

**Полезные ссылки**

* Различные манипуляции с списками <http://matthiaseisen.com/fwl/py/>

**Подсказки**

Ключевые концепции и на что стоит обращать внимание:   
циклы, for, list, tuple, slicing, indexing, numbers operations, min(), max(), sum()

**Правильный ответ**

1. import random
3. sample\_list = [random.sample(range(10), 3) for i in range(3)]
4. print sample\_list
6. sum\_first = sum(sample\_list[0][-2:])
7. min\_int\_second = min(sample\_list[1])
8. max\_int\_matrix = max(max(x) for x in sample\_list)
10. four\_index = []
11. for index, value in enumerate(sample\_list):
12. if 4 in value:
13. four\_index.append(index)
15. print "last two elements is: %s" % sum\_first
16. print "min of second row is: %s" % min\_int\_second
17. print "max of whole array is: %s" % max\_int\_matrix
18. print "index of sub-array where 4 occured is: %s" % four\_index

# Задание: Доработать программу

**Инструкции к заданию**

Есть код:

1. def main():
2. text = raw\_input("Please type yor text: ")
4. #example of count of big letters
5. big\_letters\_amount = []
6. for char in text:
7. if char >= "A" and char <= "Z":
8. big\_letters\_amount.append(char)
10. print "amount of big letters in text is %s" % len(big\_letters\_amount)
12. sentences = 0 #write code here
13. words = 0 #write code here
14. numbers = 0 #write code here
16. print "numbers: %s, words: %s, sentences: %s" % (numbers, words, sentences)
18. if \_\_name\_\_ == '\_\_main\_\_':
19. main()

Необходимо доработать программу, которая высчитывает количество цифр, слов и предложений в заданном тексте:

* программа запрашивает ввод текста
* программа анализирует строку
* и выводит статистику по тексту в виде "numbers: 9, words: 10, sentences: 10"

**Ожидаемый результат**

Python код реализующий программу

**Подсказки**

Ключевые концепции и на что стоит обращать внимание:   
string, string formatting, slicing, string indexing, list, tuple, for, bool, expressions, if, len(), string.split(), string.join(), sum(), char.isdigit()

**Правильный ответ**

1. def main():
2. text = raw\_input("Please type yor text: ")
4. # example of count of big letters
5. big\_letters\_amount = []
6. for char in text:
7. if char >= "A" and char <= "Z":
8. big\_letters\_amount.append(char)
10. print "amount of big letters in text is %s" % len(big\_letters\_amount)
12. numbers\_amount = []
13. for num in text:
14. if num >= "0" and num <= "9":
15. numbers\_amount.append(num)
17. sentences = len(text.split("."))
18. words = len(text.split(" "))
19. numbers = len(numbers\_amount)
21. print "numbers: %s, words: %s, sentences: %s" % (numbers, words, sentences)
23. if \_\_name\_\_ == '\_\_main\_\_':
24. main()

# Задание: Применение вычислений и строк

**Инструкции к заданию**

Написать программу, которая конвертирует USD в EUR по курсу (0.81) и наоборот (1.22).

Сценарий:

* программа запрашивает ввод пользователя в формате 100 USD или 100 EUR
* программа анализирует строку и разделяет число и валюту
* в зависимости от валюты, происходить конвертация введенной суммы
* после вычисления, результат показывается в форма 81 EUR and 20 cents
* если центов нет, то только 81 EUR

Пример использования, вводиться 100 USD и выводиться 81 EUR. Если есть остаток, то он выводиться как 81 EUR and 20 cents.

**Ожидаемый результат**

Рython код реализующий программу

**Подсказки**

Ключевые концепции и на что стоит обращать внимание:   
string, dictionary, dict.keys(), string formatting, numbers operations, float(), int(), round() slicing, indexing, bool, expressions, if, len(), string.split())

**Правильный ответ**

1. def main():
2. text = raw\_input("Please type yor text: ")
4. amount = float(text.split()[0])
5. currency = text.split()[1]
6. print currency
7. rates = {"USD":0.81, "EUR":1.22}
9. converted\_currency = "USD" if currency == "EUR" else "EUR"
10. to\_be\_paid = round(amount\*rates[currency],2)
11. cents = int(str(to\_be\_paid).split(".")[1])
13. if cents == 0:
14. print "%d %s" % (amount\*rates[currency], converted\_currency)
15. else:
16. print "%d %s and %d cents" % (amount\*rates[currency], converted\_currency, cents)
18. if \_\_name\_\_ == '\_\_main\_\_':
19. main()

# Задание: База данных сотрудников

**Инструкции к заданию**

Создать программу, которая определяет нужно ли выплачивать ЗП работнику:

* дано следующие данные workers(Joel, Ken, Pit), salary(Joel, Bob, Sam), contracts(Bob, Ken, Pit)
* программа запрашивает имя
* и выводит "Yes salary to be paid", если сотрудник является работником (принадлежит массиву workers), у него нету подписанного контракта (не принадлежит массиву contracts) и ему положена ЗП (принадлежит массиву salary)
* и выводит "No he shouldn't be paid salary", в остальных случаях

**Инструкции к заданию**

Python код реализующий программу

**Подсказки**

Ключевые концепции и на что стоит обращать внимание:

string, set, frozenset, if, in, set operations, string formatting

**Правильный ответ**

1. def main():
2. name = raw\_input("Please type name: ")
4. workers= {"Joel", "Ken", "Pit"}
5. salary = {"Joel", "Bob", "Sam"}
6. contracts = {"Bob", "Ken", "Pit"}
8. if name in workers & salary - contracts:
9. print "Yes salary to be paid to %r" % name
10. else:
11. print "No %r shouldn't be paid salary" % name
13. if \_\_name\_\_ == '\_\_main\_\_':
14. main()

# Задание: Расширяем кругозор

**Описание к заданию**

Найти информацию и ответить на вопросы:

* В чем разница между tuple и list?
* Как преобразовать строку и число и обратно?
* Каким модулей можно сгенирировать случайные числа?
* Зачем нужен остаток по модулю?
* Где могут применяться множества?
* Зачем нужен тип данных словари?
* Как получить отсортированный список?
* Что такое slicing? И где его можно применить?
* Что получиться в результате '---'.join(['aaa', 'bbb', 'ccc'])?
* В чем разница между изменяемыми и неизменяемыми типами?

**Подсказки**

Ключевые концепции и на что стоит обращать внимание:

tuple, list, dictionary, string, numbers, set, slicing, indexing, muttable types, immutable types

**Правильный ответ**

* В чем разница между tuple и list?  
  **Изменяемый и неизменяемый тип последовательности**
* Как преобразовать строку и число и обратно?   
  **str(), int(), float()**
* Каким модулей можно сгенирировать случайные числа?
  1. import random
  2. random.random()
* Зачем нужен остаток по модулю?   
  **Например, для того чтобы вычислять парные или непарные значения.**
* Где могут применяться множества?   
  **Например, извлечение дубликатов из последовательностей.**
* Зачем нужен тип данных словари?   
  **Ассоциативный список в котором можно хранить объекты и доступаться по ключу, а не по индексу. Словари хранятся по другому и используют другие механизмы доступа к своим элементам в отличии от индексируемых последовательностей.**
* Как получить отсортированный список?   
  **Список в котором объекты упорядоченны в порядке убывания или возрастания в зависимости от типа данных.** Можно использовать функции list.sort(), sorted()
* Что такое slicing? И где его можно применить?  
  **Это доступ к части строки или последовательности по требованию. Например, (1,2,3,4,5)[1:4:3]**
* Что получиться в результате '---'.join(['aaa', 'bbb', 'ccc'])?   
  **aaa---bbb---ccc**
* В чем разница между изменяемыми и неизменяемыми типами?   
  **Возможность изменения объекта после его создания.**