

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ № 2.6
дисциплины «Программирование на Python»

Вариант 15

Выполнил:
Кенесбаев Хилол Куат улы
2 курс, группа ИВТ-б-о-22-1,
09.03.01 «Информатика и
вычислительная техника»,
направленность (профиль)
«Программное обеспечение средств
вычислительной техники и
автоматизированных систем», очная
форма обучения

(подпись)

Руководитель практики:
Воронкин Р.А., доцент кафедры
инфокоммуникаций

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

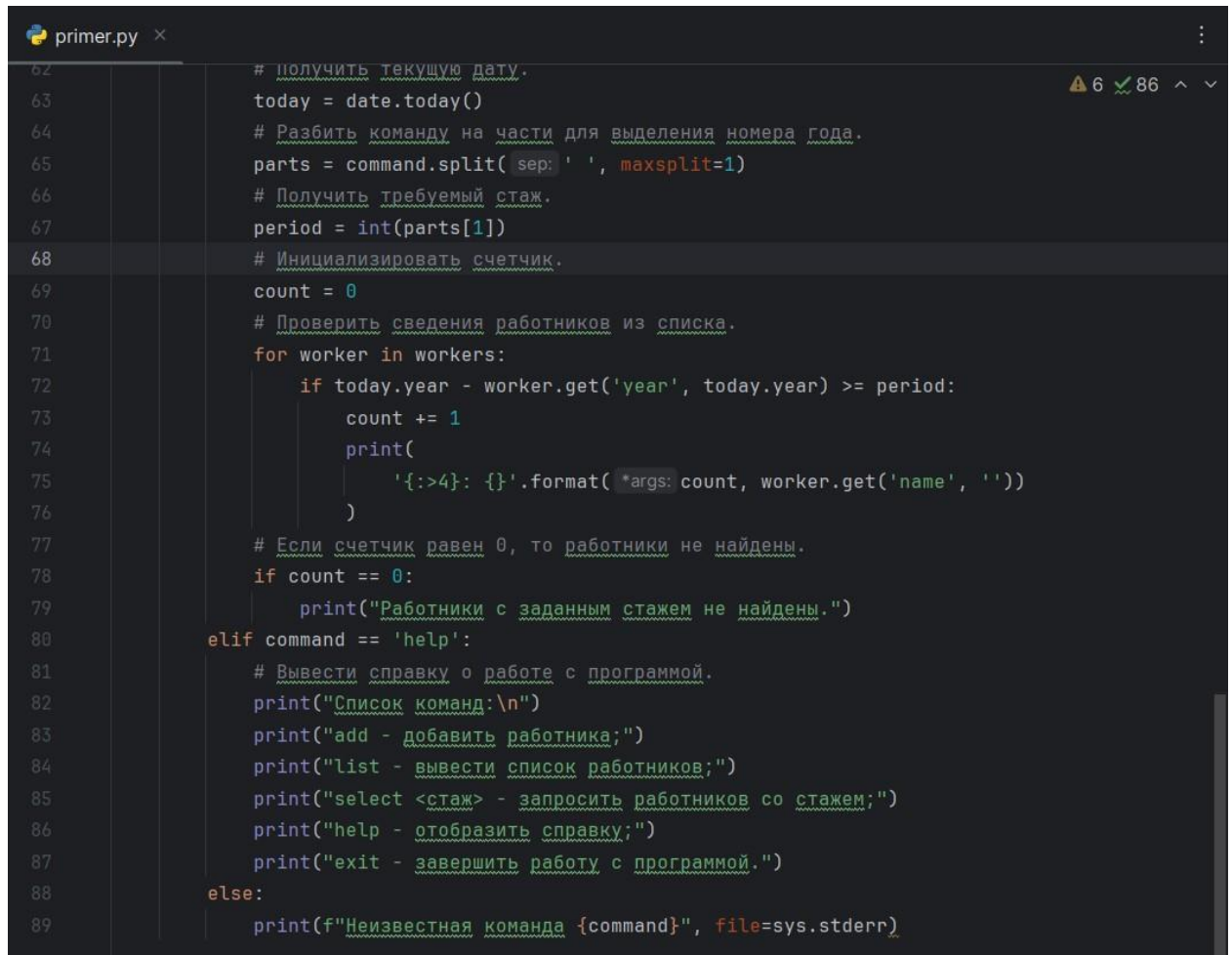
Ставрополь, 2023 г.

Тема: Работа со словарями в языке Python

Цель: приобретение навыков по работе со словарями при написании программ с помощью языка программирования Python версии 3.x.

Порядок выполнения работы:

1. Проработал примеры лабораторной работы:



```
62 # Получить текущую дату.
63 today = date.today()
64 # Разбить команду на части для выделения номера года.
65 parts = command.split( sep: ' ', maxsplit=1)
66 # Получить требуемый стаж.
67 period = int(parts[1])
68 # Инициализировать счетчик.
69 count = 0
70 # Проверить сведения работников из списка.
71 for worker in workers:
72     if today.year - worker.get('year', today.year) >= period:
73         count += 1
74         print(
75             ':{>4}: {}'.format( *args: count, worker.get('name', ''))
76         )
77 # Если счетчик равен 0, то работники не найдены.
78 if count == 0:
79     print("Работники с заданным стажем не найдены.")
80 elif command == 'help':
81     # Вывести справку о работе с программой.
82     print("Список команд:\n")
83     print("add - добавить работника;")
84     print("list - вывести список работников;")
85     print("select <стаж> - запросить работников со стажем;")
86     print("help - отобразить справку;")
87     print("exit - завершить работу с программой.")
88 else:
89     print(f"Неизвестная команда {command}", file=sys.stderr)
```

Рисунок 1 primer.py

```

add - добавить работника;
list - вывести список работников;
select <стаж> - запросить работников со стажем;
help - отобразить справку;
exit - завершить работу с программой.
>>> list
+-----+-----+-----+-----+
| № |          Ф.И.О.          |      Должность      |   Год   |
+-----+-----+-----+-----+
>>> add
Фамилия и инициалы? Кенесбаев Х.К
Должность? Студент
Год поступления? 2022
>>> add
Фамилия и инициалы? Иванов И.И
Должность? Директор
Год поступления? 2007
>>> list
+-----+-----+-----+-----+
| № |          Ф.И.О.          |      Должность      |   Год   |
+-----+-----+-----+-----+
|  1 | Иванов И.И              |      Директор      |   2007   |
+-----+-----+-----+-----+
|  2 | Кенесбаев Х.К          |      Студент      |   2022   |
+-----+-----+-----+-----+
>>> |

```

Рисунок 2 Результат работы программы primer.py

2. Решил задачу 1: Решите задачу: создайте словарь, связав его с переменной school, и наполните данными, которые бы отражали количество учащихся в разных классах (1а, 1б, 2б, 6а, 7в и т. п.). Внесите изменения в словарь согласно следующему: а) в одном из классов изменилось количество учащихся, б) в школе появился новый класс, с) в школе был расформирован (удален) другой класс. Вычислите общее количество учащихся в школе.

Код программы:

```

#!/usr/bin/env
python3# -*-
coding:
utf-8 -*-
if __name__ == "__main__":

```

```

school
= {
'1а':
25,
'1б': 27,
'2б': 20,
'6а': 30,
'7в': 22}
# а) изменилось количество учащихся в одном из
классовschool['1а'] = 26
# б) появился новый
классschool['8г'] = 18
# в) расформирован (удален) другой
классdel school['2б']
total_students = sum(school.values())
print("Общее количество учащихся в школе:", total_students)

```

```

C:\Users\HP-PC\AppData\Local\Programs\Python\Python312\python.exe C:\Users\HP-PC\PycharmProjects\lab-9\task1.py
Общее количество учащихся в школе: 123

Process finished with exit code 0

```

Рисунок 3 Результат работы программы task1.py

3. Решил задачу 2: Решите задачу: создайте словарь, где ключами являются числа, а значениями – строки. Примените к нему метод `items()`, с помощью полученного объекта `dict_items` создайте новый словарь, "обратный" исходному, т. е. ключами являются строки, а значениями – числа.

Код программы:

```

#!/usr/bin/env
python3# -*-
coding: utf-8 -*-
if __name__ == "__main__":
# Создание исходного словаря
original_dict = {1: "one", 2: "two", 3: "three"} reversed_dict = {v: k for k,
v in original_dict.items()}print("Исходный словарь:", original_dict)
print('Новый "обратный" словарь:', reversed_dict)

```

```

C:\Users\HP-PC\AppData\Local\Programs\Python\Python312\python.exe C:\Users\HP-PC\PycharmProjects\lab-9\task2.py
Исходный словарь: {1: 'one', 2: 'two', 3: 'three'}
Новый "обратный" словарь: {'one': 1, 'two': 2, 'three': 3}

Process finished with exit code 0

```

Рисунок 4 Результат работы программы task2.py

Выполнил индивидуальное задание:

Использовать словарь, содержащий следующие ключи: фамилия, имя; знак Зодиака; дата рождения (список из трех чисел). Написать программу, выполняющую следующие действия: ввод с клавиатуры данных в список, состоящий из словарей заданной структуры; записи должны быть упорядочены по датам рождения; вывод на экран информации о людях, родившихся под знаком, название которого введено с клавиатуры; если таких нет, выдать на дисплей соответствующее сообщение.

Код программы:

```
# Функция для добавления записи в список
def add_person(people_list):
    person = {}
    print("Введите данные о человеке:")
    person["фамилия"] = input("Фамилия: ")
    person["имя"] = input("Имя: ")
    person["знак Зодиака"] = input("Знак Зодиака: ")
    birthday = input("Дата рождения в формате дд.мм.гггг через точку: ").split('.')
    person["дата рождения"] = list(map(int, birthday))
    people_list.append(person)
    people_list.sort(key=lambda x: tuple(x["дата рождения"])) # Сортировка по датам
    рождения
    print("Данные о человеке добавлены.")

# Функция для вывода информации о людях по знаку Зодиака
def display_people_by_zodiac(people_list, zodiac_sign):
    found = False
    print(f"Люди с знаком Зодиака '{zodiac_sign}':")
    for person in people_list:
        if person["знак Зодиака"] == zodiac_sign:
            print(f"{person['фамилия']} {person['имя']} - {person['дата
рождения'][0]}.{person['дата рождения'][1]}.{person['дата рождения'][2]}")
            found = True
    if not found:
        print(f"Нет людей с знаком Зодиака '{zodiac_sign}'.")

# Список людей в виде словарей
people = []

# Добавление данных о людях
add_person(people)
add_person(people)

# Ввод знака Зодиака для поиска
search_zodiac = input("Введите знак Зодиака для поиска: ")
display_people_by_zodiac(people, search_zodiac)
```

```
C:\Users\HP-PC\AppData\Local\Programs\Python\Python312\python.exe C:\Users\HP-PC\Desktop\study\python\пересдача\Lab2.6\код\individualnoe.py
Введите данные о человеке:
Фамилия: Кенесбаев
Имя: Хилал
Знак Зодиака: Козерог
Дата рождения в формате дд.мм.гггг через точку: 28.12.2002
Данные о человеке добавлены.
Введите данные о человеке:
Фамилия: Кенесбаева
Имя: Гульназ
Знак Зодиака: Скорпион
Дата рождения в формате дд.мм.гггг через точку: 14.07.1998
Данные о человеке добавлены.
Введите знак Зодиака для поиска: Козерог
Люди с знаком Зодиака 'Козерог':
Кенесбаев Хилал - 28.12.2002
```

Рисунок 5 Результат работы программы individualnoe.py

Ответы на контрольные вопросы:

1. Что такое словари в языке Python?

Словарь (dict) представляет собой структуру данных (которая ещё называется ассоциативный массив), предназначенную для хранения произвольных объектов с доступом по ключу.

2. Может ли функция len() быть использована при работе со словарями?

Да может, и выведет количество ключей.

3. Какие методы обхода словарей Вам известны? Цикл for, методы items(), keys() и values().

4. Какими способами можно получить значения из словаря по ключу?

В словаре доступ к значениям осуществляется по ключам, которые заключаются в квадратные скобки или с помощью метода get().

5. Какими способами можно установить значение в словаре по ключу?

С помощью setdefault() можно добавить элемент в словарь.

помощью квадратных скобок: `my_dict['a'] = 1`

С помощью метода `update()`: `my_dict.update({'a': 1})`

6. Что такое словарь включений?

Словарь включений аналогичен списковым включениям, за исключением того, что он создаёт объект словаря вместо списка (краткая синтаксическая конструкция, предназначенная для создания словаря).

7. Самостоятельно изучите возможности функции `zip()` приведите примеры ее использования.

Функция `zip()` принимает итерируемый объект, например, список, кортеж, множество или словарь в качестве аргумента. Затем она генерирует список кортежей, которые содержат элементы из каждого объекта, переданного в функцию.

Пример:

```
keys = ['a', 'b', 'c']
values = [1, 2, 3]
my_dict = dict(zip(keys,
values))
print(my_dict)
```

Данный код выведет: `{'a': 1, 'b': 2, 'c': 3}`

8. Самостоятельно изучите возможности модуля `datetime`. Каким функционалом по работе с датой и временем обладает этот модуль?

Модуль `datetime` в Python позволяет работать с датой и временем, предоставляя классы и функции для работы с текущей датой и временем, форматирования даты и времени, а также вычисления разницы между двумя датами или временем с помощью класса `timedelta`.

- `date` – хранит дату
- `time` – хранит время
- `datetime` – хранит дату и время

Вывод: в результате выполнения работы были приобретены навыки

по работе со словарями при написании программ с помощью языка программирования Python версии 3.x.