Урок 10. Зачет.

ФИО_____

Тест:

1. *Что делает dict.fromkeys(['a','b'], [])?*

_____

2. *Что верно про set.update(iterable)?*
   A) Возвращает новое множество
   B) Изменяет множество на месте, добавляя элементы из итерируемого
   C) Удаляет дубликаты и сортирует
   D) Работает только со списками
3. *Что выведет данная программа?*

```
d = {'x': 1}
d.setdefault('x', 0)
d.setdefault('y', 2)
print(d)
```

_____

4. *Что эквивалентно записи?*

```
@decor(10)
def f(): ...
```

   A) f = decor(f)(10)
   B) f = decor(10)(f)
   C) decor = f(10)
   D) f = decor(10, f)

5. *Что верно про @staticmethod?*
   A) Автоматически получает self
   B) Не получает ни self, ни cls
   C) Можно вызывать только через экземпляр
   D) Всегда обращается к полям класса
6. *Что выведет данная программа?*

```
class X:  1 usage
    def __radd__(self, other):
        return other + 1
print(sum([1,2,3], X()))
```

_____

7. *Что верно про данный код?*

```python
class Base:  1 usage
    def __init__(self):
        self.v = 1

class Child(Base):
    def __init__(self):
        super().__init__()
        self.v += 1
```

A) Child() завершится ошибкой без super()
B) super() вызывает Base.__init__ только если явно передать self
C) Поле v создастся автоматически без вызова базового конструктора
D) У Child().v будет 2

8. *Что сделает данная программа?*

```python
class Temperature:  1 usage
    def __init__(self, c):
        self._c = c
    @property  1 usage
    def celsius(self):
        return self._c
    @celsius.setter  1 usage
    def celsius(self, value):
        if value < -273.15:
            raise ValueError("too low")
        self._c = value
t = Temperature(0)
t.celsius = -300
```

9. *Что вернет данная программа?*

```python
class Date:  1 usage
    def __init__(self, y, m, d):
        self.y, self.m, self.d = y, m, d
    @classmethod  1 usage
    def from_str(cls, s: str):
        y, m, d = map(int, s.split("-"))
        return cls(y, m, d)
class MyDate(Date):  1 usage
    pass

obj = MyDate.from_str("2025-09-02")
print(obj.y, obj.m, obj.d)
```

10. *Что вернет данная программа?*

```python
from abc import ABC, abstractmethod

class X(ABC):  3 usages
    @abstractmethod
    def f(self):
        pass
class Y(X):   2 usages
    def f(self):
        return "ok"
y = Y()
print(isinstance(y, X), issubclass(Y, X))
```

Задачи:

1. Напишите функцию, которая принимает список слов и строит словарь, где ключом является целая длина слова, а значением — список исходных слов этой длины **в порядке появления**. Пустые строки следует игнорировать. Регистр букв не менять.
   Сигнатура: def group_by_length(words: list[str]) -> dict[int, list[str]]:
   Пример: ["hi","map","to","a","see"] → {2:["hi","to"], 3:["map","see"], 1:["a"]}.
2. Сформируйте из списка целых чисел список кортежей вида (значение, сколько_раз_подряд), проходя по исходному списку слева направо. Если список пуст, верните пустой список. Нельзя пользоваться готовыми группирующими функциями стандартной библиотеки; работайте обычными циклами и сравнениями соседних элементов.
   Сигнатура: def rle(nums: list[int]) -> list[tuple[int, int]]:
   Пример: [1,1,1,2,2,5] → [(1,3),(2,2),(5,1)]
3. Создайте класс Temperature, который хранит температуру в градусах Цельсия во «внутреннем» атрибуте _c. Реализуйте свойство celsius (чтение/запись) с проверкой, что значение не ниже -273.15 (при нарушении выбрасывайте ValueError). Реализуйте свойство только для чтения fahrenheit, которое возвращает перевод в шкалу Фаренгейта по формуле C * 9/5 + 32. Изменение celsius автоматически должно менять вычисляемое значение fahrenheit.
   Пример: t=Temperature(0); t.fahrenheit → 32.0; t.celsius=-300 → ValueError.
4. Определите базовый класс Media с методом play() -> str. Создайте два подкласса: Song(title, artist) возвращает строку вида "Playing song: <title> — <artist>", и Podcast(title, host) — "Playing podcast: <title> (host:

<host>)". Напишите функцию play_all(items: list[Media]) -> list[str], которая поочерёдно вызывает play() у каждого элемента списка и возвращает список полученных строк **без** проверок типа (isinstance не использовать).
Пример: [Song("Sky","A."), Podcast("Tech","B.")] → ["Playing song: Sky — A.", "Playing podcast: Tech (host: B.)"]