Змінна a = 10 відповідає типу даних python

1. Цілому
2. Дійсному
3. Символьному
4. Булеве значення

Змінна a = 3.14 відповідає типу даних python

1. Цілому
2. Дійсному
3. Символьному
4. Булеве значення

Змінна a = “Hello Word!” відповідає типу даних python

1. Цілому
2. Дійсному
3. Символьному
4. Булеве значення

Змінна a = True відповідає типу даних python

1. Цілому
2. Дійсному
3. Символьному
4. Булеве значення

Змінна a = False відповідає типу даних python

1. Цілому
2. Дійсному
3. Символьному
4. Булеве значення

Що дозволяє зробити функція type()

1. Перевірити тип даних змінної
2. Змінити тип даних
3. Видалити змінну з неправильним типом даних
4. Оголосити змінній певний тип даних

Що з даних операторів є арифметичними

1. +, -, \*, /,
2. ==, !=, >, <
3. And, or
4. В python не має арифметичних операторів

print (7+6) який результат виконання програми

1. 7+6
2. "7 + 6"
3. ("7+6")
4. (7+6)
5. 13

print ("3+4=", 3+4) який результат виконання програми

1. 6+7= 13
2. "3+4=", 3+4
3. ("3+4=", 3+4)
4. 3+4
5. 7+6=13

Для чого призначенні умовні конструкції в python

1. Для виконання певних дій де є вибір
2. Для ітерації по обє’ктам
3. Дозволяє організовувати та зберігати дані
4. Використовується для ітерації об’єкта якщо умова є істиною

Для чого призначенні цикли в python

1. Для здійснення ітерацій по об’єктам та виконання певної дії доки певна умова є істинною
2. Для виконання певних дій де є вибір
3. Дозволяють організовувати та зберігати
4. Для виконання повторюваних дій без можливості виходу

Що з наведеного є рядком в python

1. “Hello word!”
2. {'key1': 'value1', 'key2': 'value2', 'key3': 'value3'}
3. (2, "Hello Word!", 3.0, 16, 13.14, “У”)
4. [18, 68, 69, 2, 71, 2, 0]

Яка з наведених функцій дозволяє зчитування з клавіатури

1. Input()
2. Print()
3. Type()
4. Float()

Результат виконання функції join()

1. Перетворення списка в рядок
2. Перетворення типів даних в рядок
3. Перетворює рядок у список елементів, розділених іншим рядком-розділювачем
4. Додавання елемента у список

Результат виконання функції append()

1. Додає елемент в кінець списку
2. Додає елемент в конкретну позицію
3. Додає елемент на початку списку
4. Додає елемент в середину списку

Результат виконання функції insert()

1. Додає елемент в кінець списку
2. Додає елемент в конкретну позицію
3. Додає елемент на початку списку
4. Додає елемент в середину списку

Результат виконання функції extend()

1. Для об’єднання одного списку з іншим
2. Додає елемент в кінець списку
3. Додає елемент в конкретну позицію
4. Перетворення списка в рядок

Команда elif використовується

1. Для перевірки більше однієї умови
2. Для виконання якщо всі умови є хибними
3. Для здійснення ітерацій по об’єктам та виконання певної дії доки певна умова є істинною
4. Умовний оператор для перевірки умови на хибність

Методом називають

1. Функція, що є частиною класу
2. Звичайна функція
3. Функція яка є частиною класу але нічого не повертає
4. Екземпляр класу

Що таке об’єкт в python (OOP)

1. Екземпляр класу, який містить свої власні атрибути та методи
2. Це шаблон, за допомогою якого можна створювати класи
3. Функція, що є частиною класу
4. Звичайна змінна

**Лекція 3**

Наслідування це

1. Механізм, який дозволяє створювати нові класи на основі існуючих
2. Це засіб забезпечення захисту внутрішньої реалізації класу від зовнішнього втручання
3. Це здатність об'єктів різних класів виконувати однакові дії
4. Збірки функцій та класів, які можна використовувати в програмах

Поліморфізм це

1. Механізм, який дозволяє створювати нові класи на основі існуючих
2. Це засіб забезпечення захисту внутрішньої реалізації класу від зовнішнього втручання
3. Це здатність об'єктів різних класів виконувати однакові дії
4. Збірки функцій та класів, які можна використовувати в програмах

Інкапсуляція це

1. Механізм, який дозволяє створювати нові класи на основі існуючих
2. Це засіб забезпечення захисту внутрішньої реалізації класу від зовнішнього втручання
3. Це здатність об'єктів різних класів виконувати однакові дії
4. Збірки функцій та класів, які можна використовувати в програмах

Модулі в python це

1. Механізм, який дозволяє створювати нові класи на основі існуючих
2. Це засіб забезпечення захисту внутрішньої реалізації класу від зовнішнього втручання
3. Це здатність об'єктів різних класів виконувати однакові дії
4. Збірки функцій та класів, які можна використовувати в програмах