|  |  |  |
| --- | --- | --- |
| **PYTHON. HOMEWORK\_1** | | |
| **№** | **Do** | **Done** |
| 1. | Создать переменную типа String | **x1 = str("Kek")** |
| 2. | Создать переменную типа Integer | **x2 = int(1)** |
| 3. | Создать переменную типа Float | **x3 = float(5.5)** |
| 4. | Создать переменную типа Bytes | **x4 = bytes(10)** |
| 5. | Создать переменную типа List | **x5 = [1,2,3,4]** |
| 6. | Создать переменную типа Tuple | **x6 = (1,2,3)** |
| 7. | Создать переменную типа Set | **x7 = {"Ivan", "Rinat", "Olga", "Kira"}** |
| 8. | Создать переменную типа Frozen set | **x8 = frozenset(x7)** |
| 9. | Создать переменную типа Dict | **x9 = {'dict': 1, 'dictionary': 2}** |
| 10. | Вывести в консоль все выше перечисленные переменные с добавлением типа данных. | **print(x1, type(x1))**  **print(x2, type(x2))**  **print(x3, type(x3))**  **print(x4, type(x4))**  **print(x5, type(x5))**  **print(x6, type(x6))**  **print(x7, type(x7))**  **print(x8, type(x8))**  **print(x9, type(x9))** |
| 11. | Создать 2 переменные String, создать переменную в которой сканкатенируете эти переменные. Вывести в консоль. | **x = 'kek'**  **y = 'cheburek'**  **print(x+y)** |
| 12. | Вывести в одну строку переменные типа String и Integer используя “,” (Запятую) | **x = 'kek'**  **y = int(101)**  **print(x,y)** |
| 13. | Вывести в одну строку переменные типа String и Integer используя “+” (Плюс) | **x = 'kek'**  **y = int(101)**  **print(x+str(y))** |

|  |  |  |
| --- | --- | --- |
| **PYTHON. HOMEWORK\_2** | | |
| **№** | **Do** | **Done** |
| **Arithmetic** | | |
|  | Создать переменную item\_1 типа integer. |  |
|  | Создать переменную item\_2 типа integer. |  |
|  | Создать переменную result\_sum в которой вы суммируете item\_1 и item\_2. |  |
|  | Вывести result\_sum в консоль. |  |
|  | Создать переменную result\_subtr в которой вы вычитаете большую по значению переменную из меньшей по значению. |  |
|  | Вывести result\_subtr в консоль. |  |
|  | Создать переменную result\_multi в которой вы умножаете item\_1 на item\_2. |  |
|  | Вывести result\_multi в консоль. |  |
|  | Создать переменную result\_exp в которой вы item\_1 возводите в степень item\_2. |  |
|  | Вывести result\_exp в консоль. |  |
|  | Создать переменную result\_m\_exp в которой вы item\_1 возводите в степень item\_2 используя библиотеку math. |  |
|  | Вывести result\_m\_exp в консоль. |  |
|  | Создать переменную result\_s\_root в которой вы найдёте квадратный корень любой из переменной item |  |
|  | Вывести result\_s\_root в консоль. |  |
|  | Создать переменную result\_m\_s\_root в которой вы найдёте квадратный корень любой из переменной item используя библиотеку math. |  |
|  | Вывести result\_m\_s\_root в консоль. |  |
|  | Создать переменную result\_mp\_s\_root в которой вы найдёте квадратный корень любой из переменной item используя библиотеку math используя метод pow. |  |
|  | Вывести result\_mp\_s\_root в консоль. |  |
|  | Присвоить переменной item\_1 odd значение |  |
|  | Присвоить переменной item\_2 even значение |  |
|  | Создать переменную result\_division в которой вы разделите item\_1 на item\_2. |  |
|  | Вывести result\_division в консоль. |  |
|  | Создать переменную result\_m\_floor и result\_division округлить до ближайшего целого меньшего чем result\_division. |  |
|  | Вывести result\_m\_floor в консоль. |  |
|  | Создать переменную result\_m\_ceil и result\_division округлить до ближайшего целого большего чем result\_division. |  |
|  | Вывести result\_m\_ceil в консоль. |  |
|  | Создать переменную result\_int и result\_division округлить до ближайшего целого через явное приведение. |  |
|  | Вывести result\_int в консоль. |  |
|  | Создать переменную result\_no\_division\_loss в которой вы разделите item\_1 на item\_2 без остатка. |  |
|  | Вывести result\_no\_division\_loss в консоль. |  |
|  | Создать переменную result\_division\_loss в которой вы найдёте остаток от деления item\_1 на item\_2. |  |
|  | Вывести result\_division\_loss в консоль. |  |
| **АРИФМЕТИЧЕСКИЕ ДЕЙСТВИЯ С ПРИСВАИВАНИЕМ** | | |
|  | Создать переменную item\_3 и присвоить ей целое число |  |
|  | Прибавить 10 к item\_3 с присвоением. |  |
|  | Вывести item\_3 в консоль. |  |
|  | Отнять 5 от item\_3 с присвоением. |  |
|  | Вывести item\_3 в консоль. |  |
|  | Умножить item\_3 на 3 с присвоением. |  |
|  | Вывести item\_3 в консоль. |  |
|  | Разделить item\_3 на 2 с присвоением. |  |
|  | Вывести item\_3 в консоль. |  |
|  | Возвести в степень 2 item\_3 с присвоением. |  |
|  | Вывести item\_3 в консоль. |  |
|  | Найти квадратный корень item\_3 с присвоением. |  |
|  | Вывести item\_3 в консоль. |  |
|  | Присвоить остаток от деления item\_3 |  |
|  | Вывести item\_3 в консоль. |  |
| **BOOLEAN** | | |
|  | Создать переменную b\_item\_t и присвоить True |  |
|  | Создать переменную b\_item\_f и присвоить False |  |
|  | Создать переменную b\_item\_relult\_sum и присвоить сумму b\_item\_t и b\_item\_f |  |
|  | Вывести b\_item\_relult\_sum в консоль. |  |
|  | Создать переменную b\_item\_relult\_subtr и присвоить разницу b\_item\_t и b\_item\_f |  |
|  | Вывести b\_item\_relult\_subtr в консоль. |  |
|  | Создать переменную b\_item\_relult\_multi и присвоить умножение b\_item\_t и b\_item\_f |  |
|  | Вывести b\_item\_relult\_multi в консоль. |  |
|  | Создать переменную b\_item\_relult\_division и присвоить деление b\_item\_t и b\_item\_f |  |
|  | Вывести b\_item\_relult\_division в консоль. (Получить ошибку) |  |
|  | Создать переменную b\_item\_1\_int и присвоить явное приведение b\_item\_t к int |  |
|  | Вывести b\_item\_1\_int в консоль. |  |
|  | Создать переменную b\_item\_2\_int и присвоить явное приведение b\_item\_2 к int |  |
|  | Вывести b\_item\_2\_int в консоль. |  |

|  |  |  |
| --- | --- | --- |
| **PYTHON. HOMEWORK\_3** | | |
| **№** | **Do** | **Done** |
| **Arithmetic** | | |
|  | Создать переменную int\_item со значением 10 |  |
|  | Создать переменную comp\_item со значением 18 |  |
|  | Создать переменную mult\_int в которй умножите int\_item на 2 |  |
|  | Создать переменную item\_2 со значением True |  |
|  | Создать переменную item\_3 со значением False |  |
|  | Создать переменную item\_4 со значением 0 |  |
|  | Создать переменную item\_5 со значением 1 |  |
|  | Создать переменную usd\_item со значением ‘usd’ |  |
|  | Создать переменную usd\_usd\_rate со значением 1 |  |
|  | Создать переменную eur\_item со значением ‘eur’ |  |
|  | Создать переменную usd\_eur\_rate со значением 0.86 |  |
|  | Создать переменную uah\_item со значением ‘uah’ |  |
|  | Создать переменную usd\_uah\_rate со значением 26.23 |  |
|  | Создать переменную chf\_item со значением ‘chf’ |  |
|  | Создать переменную usd\_chf\_rate со значением 0.91 |  |
|  | Создать переменную rub\_item со значением ‘rub’ |  |
|  | Создать переменную usd\_rub\_rate со значением 71.88 |  |
|  | Создать переменную byn\_item со значением ‘byn’ |  |
|  | Создать переменную usd\_byn\_rate со значением 2.46 |  |
|  | Сделать if в котором будет условие: если mult\_int больше comp\_item, то вывести в консоль (“Переменная mult\_int больше”, comp\_item) |  |
|  | Создать переменную div\_int в которй разделить int\_item на 2 |  |
|  | Сделать if в котором будет условие: если div\_int меньше comp\_item, то вывести в консоль (“Переменная div\_int меньше”, comp\_item) |  |
|  | Создать переменную item\_1 в которй прибавить 10 к переменной int\_item |  |
|  | Сделать if в котором будет условие: если item\_1 меньше comp\_item, то вывести в консоль (“Переменная item\_1 меньше”, comp\_item), иначе, вывести в консоль (“Переменная item\_1 больше или равна”, comp\_item) |  |
|  | Сделать if в котором будет условие: если item\_2, то вывести в консоль (“Переменная item\_2 = ”, item\_2), иначе, вывести в консоль (“Переменная item\_2 = ”, item\_3) |  |
|  | Сделать if в котором будет условие: если item\_3, то вывести в консоль (“Переменная item\_3 = ”, item\_2), иначе, вывести в консоль (“Переменная item\_3 = ”, item\_3) |  |
|  | Сделать if в котором будет условие: если item\_5, то вывести в консоль (“Переменная item\_5 = ”, item\_5), иначе, вывести в консоль (“Переменная item\_5 = ”, item\_4) |  |
|  | Сделать if в котором будет условие: если item\_4, то вывести в консоль (“Переменная item\_4 = ”, item\_5), иначе, вывести в консоль (“Переменная item\_4 = ”, item\_4) |  |
|  | Создать переменную currency\_convertor со значением item\_2 |  |
|  | Сделать if в котором будет условие: если currency\_convertor, то выполнять следующие шаги задания, иначе, вывести в консоль (“Переменная currency\_convertor = ”, item\_3) |  |
|  | Внутри if currency\_convertor сделать следующие If условия : |  |
| 31.1 | Создать переменную currency\_usd со значением usd\_item |  |
| 31.2 | Создать переменную target\_currency со значением eur\_item |  |
| 31.3 | Создать переменную target\_currency\_amount значением 50 |  |
| 31.4 | Создать переменную currency\_result со значением 0 |  |
| 31.5 | Сделать if в котором будет условие: если target\_currency равен ‘eur’, то в теле этого if в значении переменной currency\_result высчитать сколько долларов получится при target\_currency\_amount и usd\_eur\_rate. Результат вывести в консоль (target\_currency\_amount, eur\_item, “=”, currency\_result, usd\_item) |  |
| 31.6 | Сделать elif в котором будет условие: если target\_currency равен ‘uah’, то в теле этого if в значении переменной currency\_result высчитать сколько долларов получится при target\_currency\_amount и usd\_uah\_rate. Результат вывести в консоль (target\_currency\_amount, uah\_item, “=”, currency\_result, uah\_item) |  |
| 31.7 | Сделать elif с остальными валютами |  |
| 31.8 | Последним оставить else, при выполнений которого в консоль выведется (“Unknow currency”) |  |